
TECHNICKÁ UNIVERZITA V LIBERCI

Fakulta mechatroniky, informatiky a mezioborových studií

Studijní program: N 2612 – Elektrotechnika a informatika

Studijní obor: 3902T005 – Automatické řízení a inženýrská informatika

Numerická simulace obtékání kmitajícího tělesa

CFD simulation of flow past a vibrating body

Diplomová práce

Autor: **Bc. Jiří Staněk**

Vedoucí práce: Ing. Petr Šidlof, Ph.D.

V Liberci 12. 5. 2011

Prohlášení

Byl(a) jsem seznámen(a) s tím, že na mou diplomovou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci (TUL) nezasahuje do mých autorských práv užitím mé diplomové práce pro vnitřní potřebu TUL.

Užiji-li diplomovou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti TUL; v tomto případě má TUL právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Diplomovou práci jsem vypracoval(a) samostatně s použitím uvedené literatury a na základě konzultací s vedoucím diplomové práce a konzultantem.

V Liberci 12. 5. 2011

Podpis

Poděkování

Děkuji především vedoucímu mé diplomové práce Ing. Petru Šidlofovi, Ph.D. za ochotu a vstřícnost při vzniku této diplomové práce a s tím spjatém řešení problémů.

Práce byla částečně podpořena z projektu GAČR P101/11/0207 "Coupled problems of fluid and solid mechanics - nonlinear aeroelasticity".

Abstrakt

V této diplomové práci byla řešena kompletní simulace obtékání statického a kmitajícího tělesa nestlačitelnou tekutinou v uzavřeném kanálu. Na tvorbu geometrie a sítě byl použit program Gmsh, další část byla realizována pomocí programového balíku OpenFoam a vizualizace byla provedena programem ParaView. U obtékání statické překážky byl kladen důraz na verifikaci modelu, která určila, nakolik jsou výsledná vypočtená data vyhovující v porovnání s výsledky jiných vědeckých skupin. V případě obtékání dynamické překážky byl řešen problém dynamické sítě, kde bylo nutno pomocí parametru difuzivita zajistit její přípustnou deformaci.

Verifikace výpočetního modelu provedená na případu obtékání statického válce prokázala velmi dobrou shodu výsledků CFD simulace v programu OpenFOAM s benchmarkovými daty. Po provedení verifikace byla realizována simulace proudění kolem pohyblivé překážky na dvou příkladech - 2D kmitající válec v přímém kanálu a kmitající profil lidských hlasivek.

Klíčová slova: obtékání těles, Navier – Stokesovy rovnice, dynamická síť, laminární simulace proudění.

Abstract

This diploma thesis was solving the complete simulation of flow past a static and oscillating body by incompressible fluid in a closed duct. We have used the Gmsh program for the geometry and mesh generation, the next part has been realized by means of the OpenFoam package and the visualization has been carried out by the ParaView program. Concerning the flow past static body we have placed emphasis on the model verification which determined to what extent the resulting calculated data are satisfactory in comparison with the results of other scientific teams. In the case of flow past dynamic body we were dealing with the problem of dynamic mesh, where it was necessary to ensure its acceptable deformation by means of the diffusivity parameter.

Verification of the analysis model, which was carried out on the case of flow past static cylinder, proved a very good result correspondence between the CFD simulation in the OpenFoam program and the benchmark data. After the verification process there has been realized a simulation of flow past dynamic body on two

examples – 2D oscillating cylinder in a direct duct and oscillating profile of human vocal folds.

Keywords: flow past bodies, Navier – Stokes equations, dynamic mesh, simulation of laminar flow.

Obsah

Prohlášení.....	3
Poděkování.....	4
Abstrakt.....	5
Obsah	7
1 Úvod.....	9
1.1 Cíl práce	9
1.2 Fyzika proudění.....	9
1.2.1 Viskozita	11
1.2.2 Reynoldsovo číslo.....	12
1.3 Vliv Reynoldsova čísla na charakter proudění.....	12
1.4 Navier – Stokesovy rovnice	14
1.4.1 Numerické řešení Navier – Stokesových rovnic.....	15
1.5 Vírová řada.....	16
2 Obtékání statického válce v kanálu	17
2.1 Generování sítě.....	17
2.2 Výpočet metodou konečných objemů	18
2.2.1 OpenFoam.....	18
2.2.2 Nastavení parametrů pro výpočet proudění kolem válce v kanálu	20
2.3 Vizualizace výsledků v programu ParaView	26
2.4 Ověření výsledků simulace s výsledky testovací úlohy	27
2.4.1 Validace a verifikace CFD modelu.....	27
2.4.2 Vyhodnocení výsledků simulace proudění kolem válce v kanálu.....	28
2.4.3 Porovnání rychlostních profilů	40
2.5 Vizualizace proudění kolem statického válce v kanálu	42
3 Obtékání kmitající překážky	44
3.1 Dynamická síť	44
3.1.1 Pevná topologie.....	44
3.1.2 Proměnná topologie	46
3.2 Kmitající válec v kanálu.....	47
3.2.1 Nastavení numerické simulace	47

3.2.2	Výsledky numerické simulace	51
3.3	Kmitající profil hlasivek	54
3.3.1	Zjednodušený model hlasivek	54
3.3.2	Model hlasivek M5	56
4	Závěr	61
	Použitá literatura	63
	Přílohy.....	65
	Soubor boundary: popis hranic objektu	65
	Soubor turbulenceProperties: charakter proudění.....	66
	Soubor transportProperties: fyzikální konstanty média.....	67
	Soubor fvSchemes: numerická schémata metody konečných objemů	68
	Soubor fvSolution: lineární řešiče	70
	Soubor controlDict: parametry simulace	72
	Soubor U: okrajové podmínky pro rychlost	74
	Soubor p: okrajové podmínky pro tlak	75
	Soubor dynamicMeshDict: deformace sítě - difusivita	76
	Soubor pointMotionU: okrajové podmínky pro pohyb sítě.....	77

1 Úvod

1.1 Cíl práce

V diplomové práci byla řešena problematika matematického modelování obtékání tělesa tekutinou v uzavřeném kanálu. Byla vytvořena geometrie a síť pomocí programu Gmsh. Na výpočet metodou konečných objemů byl použit program OpenFoam a na vyhodnocení výsledků simulace byl použit program ParaView.

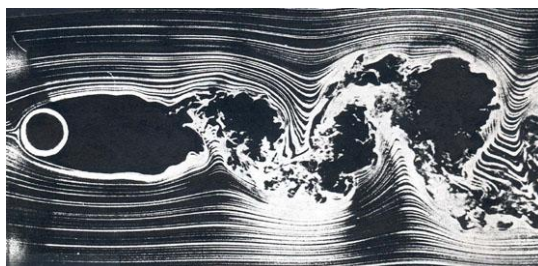
Další částí byla validace nasimulovaného proudění kolem válce v uzavřeném kanálu oproti testovací (benchmarkové) úloze, pomocí které bylo stanoveno, nakolik se naše simulace liší od skutečně správných výsledků. Simulace byla posuzována z hlediska tlakové difference a rychlostních průběhů podél zvolených křivek. V dostupných testovacích úlohách byla k dispozici jak vypočtená data, tak data získaná experimentálně.

Poslední částí diplomové práce byly simulace obtékání kmitajícího válce v kanálu a modelu lidských hlasivek. Zde bylo nutné řešit problém s konvergencí výpočtu vlivem příslušné deformace sítě. Byly vytvořeny dva modely lidských hlasivek, kde první z nich měl velmi jednoduchou geometrii oproti druhému, ve kterém byl již použit konkrétní model M5, který se reálně používá a má přesně definovanou geometrii uvedenou v práci [14].

V dnešní době zažívá matematické modelování, respektive simulování konkrétních fyzikálních problémů, velký rozmach. Je to logické, jelikož matematické modelování je efektivní cesta k odlaďování problémů při návrhu různých zařízení. Jako příklad lze uvést ladění aerodynamiky aut, či letadel. Matematické modelování se rozvíjí také proto, že máme k dispozici dostatečný výpočetní výkon pro jeho řešení. Stále ovšem neexistují natolik výkonné počítače, s nimiž bychom mohli řešit problémy velkého rozsahu (např. atmosférické proudění) s dostatečnou přesností.

1.2 Fyzika proudění

V diplomové práci byla řešena simulace proudění kolem válce v uzavřeném kanálu (obr. 1).

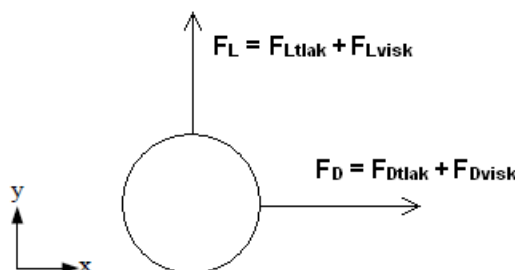


Obr. 1: Výsledek experimentu obtékání válce v kanálu s vizualizací proudnic pomocí barviva. Převzato z [10].

V kanálu proudí viskózní tekutina, tj. tekutina, která se vyznačuje vnitřním třením mezi jednotlivými vrstvami. Na stěnách kanálu i válce je nulová rychlost. Rychlostní profil proudění v kanálu bez překážky se blíží parabolickému profilu. Samozřejmě lze tvrdit, že rychlost proudění je v užších místech kolem válce vyšší, nežli v místech mimo překážku. V místě za válcem se tvoří úplav s nižším tlakem, jehož velikost závisí na Reynoldsově čísle.

Obecně na válec působí dva druhy sil, kterými jsou tlaková a viskózní síla. Každá tato síla se dá rozložit na 3 složky podle svých příspěvků do jednotlivých souřadných směrů. Viskózní síla představuje snahu válce nechat se „unést“ vlivem tření kapaliny o povrch válce. Zde hraje roli viskozita tekutiny. Je jasné, že pokud bude válec obtékat med, tak viskózní složka bude větší, nežli když jej bude obtékat voda. Tlaková síla působí na válec na základě rozdílu tlaků před i za válcem. Při obtékání překážky reálnou kapalinou je za tělesem místo s nižším tlakem nežli před tělesem, tudíž zde působí tlaková síla, která má tendenci „unést“ překážku ve směru proudění.

Síly, které se dají odvodit z tlakové a viskózní síly jsou odporová (F_D) a vztlaková síla (F_L), zobrazené na obr. 2. Odporová síla je dána součtem složek tlakové (F_{Dtlak}) a viskózní (F_{Dvisk}) síly ve směru proudění. Její směr je totiž stejný se směrem rychlosti proudění a její velikost závisí mimo jiné na tvaru překážky. Naproti tomu vztlaková síla je dána součtem složek tlakové (F_{Ltlak}) a viskózní (F_{Lvisk}) síly kolmo ke směru proudění.



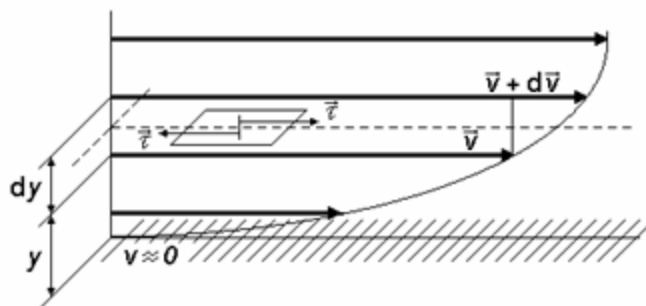
Obr. 2: Vztlaková a odporová síla působící na válec

1.2.1 Viskozita

Viskozita je fyzikální konstanta charakterizující tekutinu (podobně jako hustota, či permitivita). V tekutině jsou mezní vrstvy pohybující se rozdílnými rychlostmi. Jednotlivé mezní vrstvy se o sebe třou, a tudíž zde působí třecí síly, které působí proti toku tekutiny. V důsledku rozdílných rychlostí a třecích sil mezních vrstev vzniká v tekutině tečné napětí $\vec{\tau}$:

$$\vec{\tau} = \eta \frac{d\vec{v}}{dy}, \quad [\text{kg} \cdot \text{m}^{-1} \cdot \text{s}^{-1}]. \quad (1)$$

Kde η je koeficient úměrnosti s názvem dynamická viskozita a poměr $\frac{d\vec{v}}{dy}$ je gradient rychlosti. Vektorové veličiny budou v diplomové práci značeny tučně nebo klasicky se šipkou ($\mathbf{v} = \vec{v}$). Tečné napětí je znázorněno na obr. 3. Jednotkou dynamické viskozity je $\text{N} \cdot \text{s} \cdot \text{m}^{-2}$.



Obr. 3: Tečné napětí působící mezi vrstvami tekutiny s různou rychlostí.

Převzato z [15].

Při výpočtech proudění tekutin se často počítá s kinematickou viskozitou ν :

$$\nu = \frac{\eta}{\rho}, \quad [\text{m}^2 \cdot \text{s}^{-1}]. \quad (2)$$

Kde ρ je hustota tekutiny. Pro vzduch se v tabulkách udává hodnota kinematické viskozity $1.58 \cdot 10^{-5} \text{ m}^2 \text{ s}^{-1}$. [15]

1.2.2 Reynoldsovo číslo

Jedním ze základních parametrů charakterizujících proudění je Reynoldsovo číslo Re :

$$Re = \frac{d \cdot |U|}{\nu}, \quad (3)$$

kde d je charakteristický rozměr (například průměr válce), $|U|$ je charakteristická rychlost proudění a ν je kinematická viskozita.

Jedná se o bezrozměrné číslo udávající poměr setrvačných sil vůči viskozním silám při daných parametrech proudění. Velikost Reynoldsova čísla nám určuje, zda bude proudění spíše laminární, či turbulentní. Pro malá Reynoldsova čísla se bude jednat o laminární proudění. Kdežto pro velká Re se bude jednat spíše o turbulentní proudění. Pro proudění v uzavřených kanálech dochází k přechodu k turbulenci pro $Re = 2000 - 4000$. Konkrétně v uzavřených kanálech, jako je uzavřená trubka, dochází k přechodu k turbulenci pro $Re = 2300$. Tato hodnota bývá označována jako kritická hodnota Reynoldsova čísla.

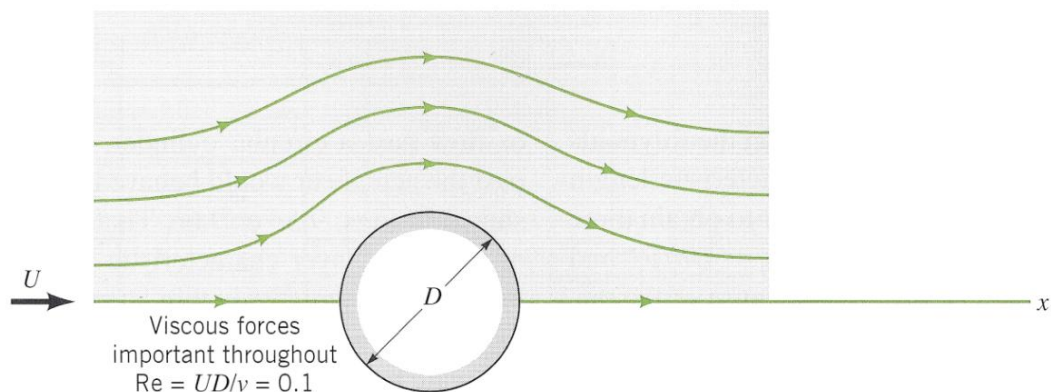
Ze vzorce (3) je vidět, že pokud máme velké rozměry překážky či velkou rychlost nebo malou viskozitu tekutiny, tak Re bude také velké a proudění bude mít turbulentní charakter. To znamená, že bude převažovat vliv setrvačných sil. Naopak pro malé rozměry překážek, malou rychlost nebo velkou viskozitu tekutiny, bude Re malé a proudění bude laminární, což znamená, že bude převažovat vliv viskozních sil.

1.3 Vliv Reynoldsova čísla na charakter proudění

Na charakteru proudění se významnou měrou podílí Reynoldsovo číslo. Pro různá Reynoldsova čísla dostaneme poněkud odlišné chování proudění tekutiny v uzavřeném kanálu. Pokud bude do uzavřeného kanálu umístěna překážka, v konkrétním případě válec, tak nás bude především zajímat chování tekutiny za válcem, kde můžeme sledovat vznik úplavu a odtržení proudnic.

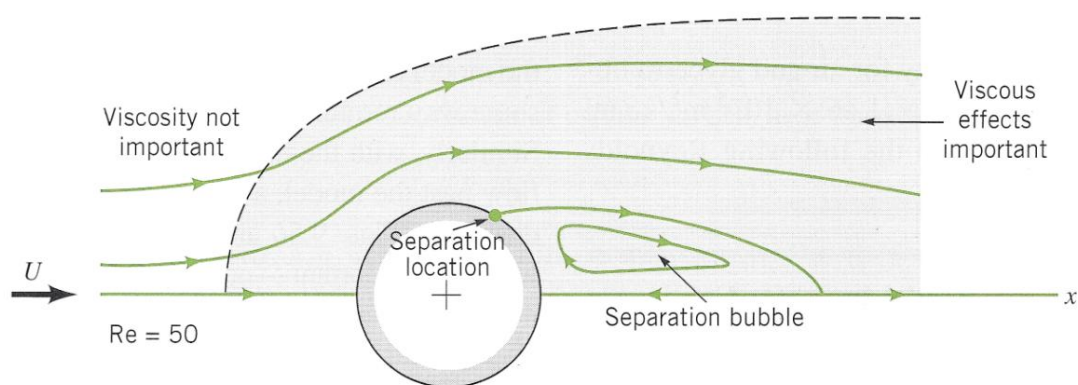
Pro malá Reynoldsova čísla ($Re = 0,1$) má viskózní efekt převahu ve všech směrech kolem válce a také ve velké části kanálu. Při porovnání průběhu proudnic před i za válcem je vidět, že tyto průběhy jsou stejné, tudíž nedochází k jejich odtržení

a vzniku úplavu. Pro názornost je charakter proudění pro $Re = 0,1$ zobrazen na obr. 4. [4]



Obr. 4: Charakter proudění pro $Re = 0,1$. Převzato z [4].

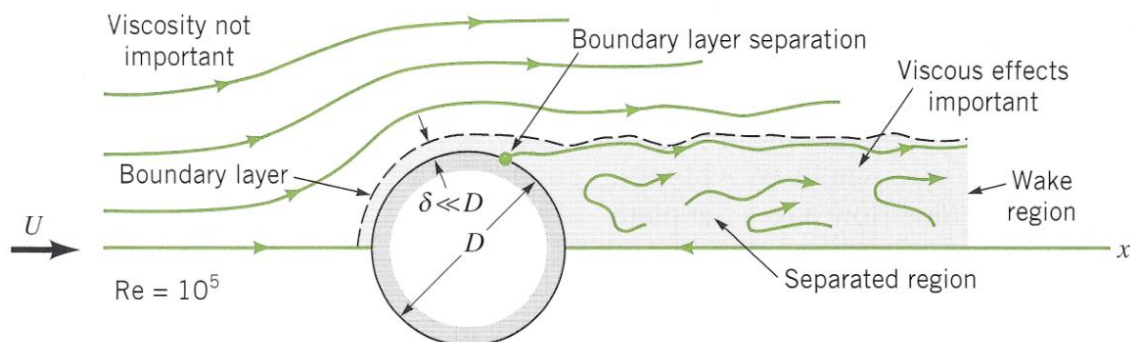
Pro vyšší Reynoldsova čísla ($Re = 50$) se před a okolo válce zmenšila oblast, kde dominuje viskózní efekt. Nicméně tato oblast zůstala stejně velká za válcem. Tímto rozdílem se ztratí symetrie charakteru proudění. Při vyšším Reynoldsově čísle začne v proudění dominovat také další důležitá vlastnost, kterou je setrvačnost proudící tekutiny. V určitém místě dojde k odtržení proudnic od válce, což je způsobeno obráceným gradientem tlaku. Tím pádem tekutina nekopíruje celý povrch válce. To znamená, že k „odtržení“ proudění od válce dojde dříve, než stačí proudění obtéci celou válcovou překážku. Za válcem vznikne úplav, kde se část proudění vrací k válci proti směru hlavního proudění. Příklad, kde je zobrazen charakter proudění pro $Re = 50$, máme na obr. 5. [4]



Obr. 5: Charakter proudění pro $Re = 50$. Převzato z [4].

Při ještě vyšším Reynoldsově čísle ($Re = 10^5$) se vrstva, kde dominuje viskózní efekt, velmi zúží a téměř obepíná válec v jeho přední části. Zatímco v jeho zadní části vzniká tak zvaný turbulentní region. Mimo mezní vrstvu kolem válce a turbulentní

oblast se proudění chová jako neviskózní. Samozřejmě viskozita tekutiny je ve všech místech kanálu stejná, nicméně, zda považujeme viskózní efekt za důležitý, či nikoliv, záleží na oblasti kanálu, kterou bereme v potaz. Uvnitř mezní vrstvy a turbulentní oblasti je mnohem větší gradient rychlosti nežli mimo tyto oblasti. Charakter proudění pro $Re = 10^5$ je znázorněn na obr. 6. [4]



Obr. 6: Charakter proudění pro $Re = 10^5$. Převzato z [4].

1.4 Navier – Stokesovy rovnice

Tyto rovnice vznikly z druhého Newtonova zákona a byly pojmenovány podle francouzského matematika L. M. Naviera (1785 - 1836) a Sira G. G. Stokesa (1819 - 1903). Vektorový tvar Navier – Stokesových rovnic pro nestlačitelné tekutiny je

$$\rho \left(\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} \right) = -\nabla p + \eta \nabla^2 \mathbf{v} + \mathbf{S}_M, \quad (4)$$

kde \mathbf{v} je vektor rychlosti proudění, ρ je hustota tekutiny, p je tlak, η je dynamická viskozita a \mathbf{S}_M jsou externí síly. Levá strana rovnice reprezentuje setrvačnost (zrychlení) proudění a na straně pravé se vyskytují vztahy pro sílu. Fakt, že při proudění tekutina nevzniká ani nezaniká, je vyjádřen rovnicí kontinuity pro nestlačitelné tekutiny

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0, \quad (5)$$

kde u, v, w jsou příspěvky rychlosti do souřadných směrů. Tím získáme čtyři rovnice pro čtyři proměnné (u, v, w, p). [11]

Rovnice jsou používány při řešení problémů spjatých s prouděním Newtonovské tekutiny, což znamená, že existuje závislost mezi tečným napětím ($\vec{\tau}$)

a gradientem rychlosti popsaná vztahem (1). Newtonovské tekutiny jsou popsány základní materiálovou konstantou s názvem dynamická viskozita (η). Konkrétní použití rovnic může být při řešení obtékání vzduchu kolem křídel letadel, proudění tekutiny v potrubí, atd.

Výsledkem řešení Navier – Stokesových rovnic je velikost rychlosti rozložená podle příspěvků do každého směru v souřadném systému, v každém čase a bodě, v celé tekutině. Tomuto výsledku se říká proudové nebo rychlostní pole.

Bohužel Navier – Stokesovy rovnice jsou nelineární, parciální, diferenciální rovnice druhého řádu. Ve většině reálných případů nelze nalézt analytické řešení. Nelinearita je způsobena konvektivním členem. Numerické řešení Navier - Stokesových rovnic je velmi obtížné, nicméně, pokud by šly rovnice stabilně (s časem konvergujeme k výsledku) vyřešit, měli bychom k dispozici úplný popis chování proudění, včetně turbulencí. Bohužel by na to byla potřeba velmi jemná síť, kterou není schopen vyřešit pro střední a vysoká Reynoldsova čísla ani ten nejvýkonnější superpočítač na světě. Této metodě se říká Direct numeric simulation (DNS).

Abychom docílili použitelnosti rovnic na dostupných počítačích, tak se počítá v rovnicích s vyprůměrovaným rychlostním polem, obsahujícím model pro turbulence. Této metodě se říká Reynolds – averaged Navier – Stokes equation (RANS). Bohužel je zde použit model pro turbulence, takže se od reálného chování proudění opět vzdalujeme. Tato metoda je v praxi využívána zejména v komerčních CFD řešičích (Computational Fluid Dynamics – výpočet proudění tekutin).

Další možností, jak Navier – Stokesovy rovnice v dnešní době aplikovat, je metoda, která je svou výpočetní náročností mezi DNS a RANS. Jmenuje se Large – eddy simulation (LES). [4] [12]

1.4.1 Numerické řešení Navier – Stokesových rovnic

Pro numerické řešení konkrétních problémů jsou k dispozici tři základní metody:

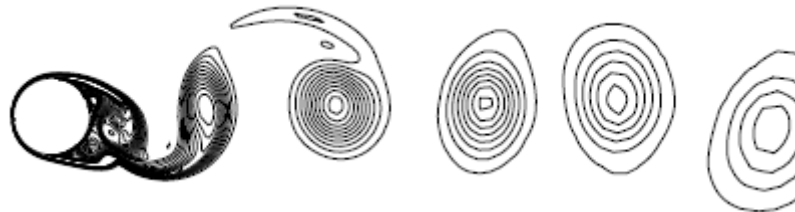
- Metoda konečných diferencí (Finite difference method)
 - Používá se pro jednoduché geometrie se strukturovanou sítí.
- Metoda konečných objemů (Finite volume method)
 - Nalezneme ji v komerčních CFD kódech (Ansys, Fluent)

- Metoda konečných prvků (Finite element method)
 - Metoda vyššího řádu, která dává při stejně jemné síti přesnější výsledky, než metoda konečných objemů. Pro vyšší Reynoldsova čísla je obvykle nutné výpočet uměle stabilizovat.

1.5 Vírová řada

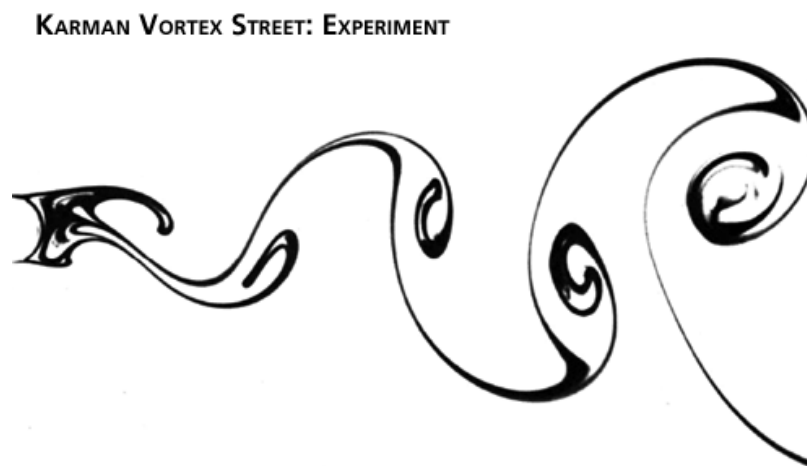
Při modelování obtékání válce v kanálu je dobré se seznámit s konkrétní teorií a vědět, jak by se proudění mělo chovat. Tento problém je popsán například v práci [5].

Jak lze v práci [5] vyčíst, tak se za válcem v kanálu začne tvořit vírová řada. Charakter odtrhávání vírů a jejich frekvence závisí na Reynoldsově číslu. Výsledek URANS 2D (Unsteady Reynolds-averaged Navier-Stokes) simulace je vidět na obr. 7, kde jsou zobrazeny isočáry vírovosti. Toto proudění mělo Reynoldsovo číslo 3900.



Obr. 7: URANS simulace pro 2D případ. Převzato z [5].

Výsledek experimentu, kdy se tvoří vírová řada, je vidět na obr. 8.



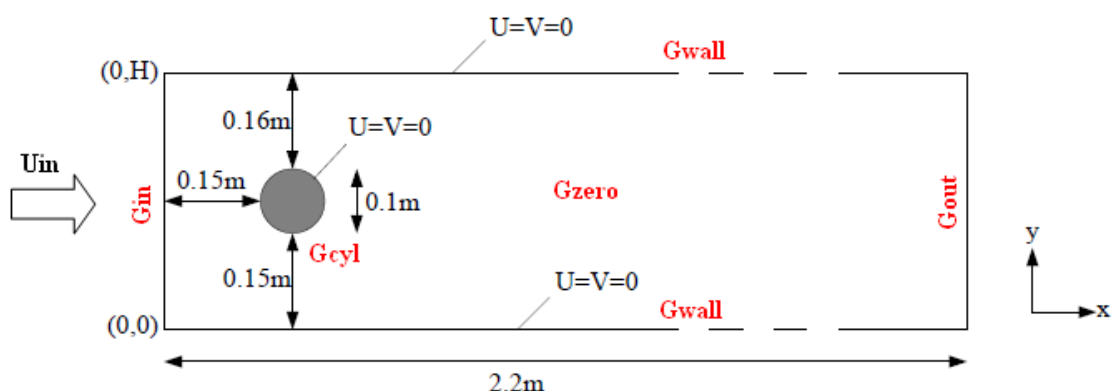
Obr. 8: Vírová řada. Převzato z [7].

2 Obtékání statického válce v kanálu

V úvodní části byla řešena obecná teorie proudění. Teoretické poznatky budou nadále využity při modelování konkrétního problému, kterým je obtékání statického válce v kanálu, a to jak při nastavování simulací, tak při vyhodnocení jejich výsledků.

2.1 Generování sítě

Prvním krokem při modelování proudění kolem válce v uzavřeném kanálu je tvorba jeho geometrie a sítě. Geometrie válce je definována na obr. 9. Rozměry byly převzaty z vědecké práce [9] a to proto, aby bylo možné vypočtená data porovnat s daty uvedenými ve vědecké práci [9], kde jsou zaznamenány právě přesné rozměry geometrie a parametry proudění testovací úlohy, jakou je válec v kanálu, včetně vypočtených i naměřených výsledků. Na obr. 9 jsou červeně napsány názvy jednotlivých hranic. Jak se bude proudění na hranici chovat definují okrajové podmínky.



Obr. 9: Geometrie válce v kanálu. Rozměry brány dle testovací úlohy [9].

K tvorbě geometrie a sítě byl vybrán program Gmsh, který je volně stažitelný na internetu. Program Gmsh je dostupný jak pro platformu Windows, tak Linux. Program Gmsh umožňuje tvorbu strukturovaných i nestrukturovaných sítí. Oproti tomu program BlockMesh, který je implementován v balíku OpenFoam, umožňuje pouze tvorbu strukturovaných sítí. Základy práce s programem GMSH jsou popsány v manuálu [3].

Modelování proudění kolem válce bylo řešeno jako 2D úloha s tím, že OpenFoam přistupuje k řešení těchto úloh tak, že do OpenFoamu musejí být

importovány „3D“ sítě o tloušťce 1 elementu a při samotném výpočtu není souřadnice z brána v potaz.

Síť vytvořenou pomocí programu Gmsh je třeba konvertovat do formátu OpenFoam příkazem

gmshToFoam xxxx.msh.

2.2 Výpočet metodou konečných objemů

V tomto odstavci budou popsány jednotlivé druhy okrajových podmínek a bude provedeno jejich konkrétní nastavení. Budou zde nastaveny nezbytné parametry simulace, jako je například časový krok. Celý výpočet bude realizován pomocí programu OpenFoam.

2.2.1 OpenFoam

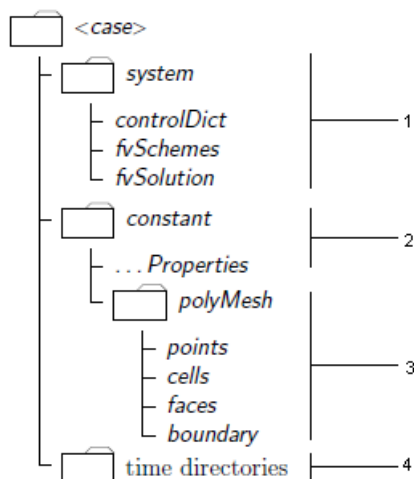
Programový balík OpenFoam je opensource balík, udržovaný v rámci GNU-GPL (General Public Licence) a využívající se primárně na numerické výpočty proudění stlačitelných i nestlačitelných tekutin. Nicméně existují i řešiče (solvers) na vedení tepla, elektromagnetismus, elasticitu, spalování, finance atp. Pro výpočty nestlačitelného proudění, které je v diplomové práci řešeno, jsou k dispozici například řešiče uvedené v tab. 1.

<i>icoFoam</i>	Řešič laminárního proudění nestlačitelné Newtonovské tekutiny
<i>pisoFoam</i>	Řešič proudění nestlačitelných tekutin
<i>simpleFoam</i>	Řešič ustáleného stavu turbulentního proudění nestlačitelných tekutin
<i>pimpleFoam</i>	Vznikl spojením řešičů PISO + SIMPLE s tím, že dovoluje nastavit delší časový krok simulace
<i>pimpleDyMFoam</i>	Využívá řešič pimple, ale je použitelný na dynamické sítě

Tab. 1: Příklady řešičů výpočtu proudění nestlačitelných tekutin

Do zvláštní kategorie řešičů pro výpočet proudění patří *potentialFoam*, který se používá pro výpočet počátečního proudového pole pro Navier – Stokesovy rovnice. Na výpočet proudění kolem válce v kanálu byl použit řešič *pimpleFoam*. [1]

Celý program OpenFoam pracuje na základě úprav jednotlivých textových souborů. Z tohoto důvodu je nutné znát adresářovou strukturu (obr. 10), abychom se snadno orientovali a věděli, kde se nachází příslušný soubor, jež chceme upravovat.



Obr. 10: Adresářová struktura programu OpenFoam. Převzato z [1].

Popis jednotlivých adresářů:

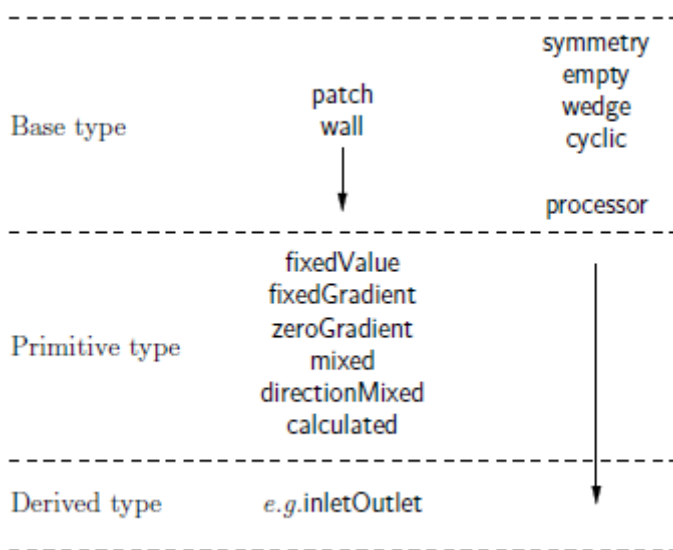
- 1) V adresáři **system** se nastavují vlastnosti výpočtu, jako je například: délka časového kroku, numerické metody pro řešení diferenciálních rovnic, délka výpočtu, atp..
- 2) V adresáři **constant** se nacházejí vlastnosti prostředí, což samozřejmě záleží na tom, co právě modelujeme. Pro modelování tekutin je charakteristickou veličinou například viskozita.
- 3) **polyMesh** je podadresářem adresáře constant a obsahuje informace o geometrii a síti modelovaného objektu.
- 4) Tyto adresáře (**time directories**) obsahují výstupní soubory simulace v námi zvolených časových krocích. Název adresáře odpovídá konkrétnímu času, ve kterém jsou vypočítané hodnoty simulace. Na začátku je nutné definovat počáteční podmínky v konkrétním čase, od kterého se budou vypočítávat další hodnoty. [1]

2.2.2 Nastavení parametrů pro výpočet proudění kolem válce v kanálu

Tento odstavec bude popisovat nastavení důležitých parametrů pro samotný výpočet. V diplomové práci byl zvolen laminární model simulace. V OpenFoamu se značí pod pojmem *laminar*.

Bylo potřeba nasimulovat obtékání válce v kanálu (obr. 9). Konkrétně bude zajímavé sledovat chování proudění za válcem, jelikož před ním se nic zajímavého dít nebude. Těmto dobře popsáním (jak měřením, tak výpočty) případům, se říká testovací (benchmarkové) úlohy.

Nyní bude zmíněno nastavení okrajových podmínek. Na obr. 11 je vidět základní rozdělení okrajových podmínek. V nejvyšší vrstvě (Base type) jsou typy hranic objektu na základní úrovni geometrie, což znamená, že se aplikují do souboru *boundary*. Prostřední vrstvou (primitive type) určujeme okrajové podmínky vlastního výpočtu. Tyto podmínky se používají v časovém adresáři společně s nastavením počátečních podmínek.



Obr. 11: Dělení okrajových podmínek. Převzato z [1].

Veškeré nastavení okrajových podmínek je nutné provést na základě textu [9], aby bylo možné dále výsledky korektně porovnat.

Nastavení typu okrajových podmínek v souboru **boundary** [Příloha 1]:

Gcyl, Gin, Gout, Gwall	– patch
Gzero	– empty

Typy hranic objektu na základní úrovni geometrie:

- patch** - Obecná podmínka, na kterou lze aplikovat velké množství okrajových podmínek týkajících se samotného výpočtu
- wall** - Charakterizuje nejlépe vlastnosti chování kapaliny v blízkosti stěny. Vzhledem k charakteristickému profilu proudění modeluje mezní vrstvu, ve které dochází k velké změně rychlosti proudění (velký gradient rychlosti) pomocí stěnových funkcí. Rychlost zde roste v malé oblasti z nuly na námi definovanou rychlost.
- symmetryPlane** - Simuluje rovinu symetrie (proudění v otevřeném prostoru).
- wedge** - Používá se u axisymetrických geometrií. Stačí definovat pouze pětistupňový výřez celého objemu s tloušťkou jednoho prvku sítě. Tuto podmínku lze aplikovat na stranu výřezu.
- empty** - Používá se při modelování ve 2D pro část hranice s normálou do třetího rozměru. [1]

Do složky constant se musí vytvořit soubory **turbulenceProperties** [Příloha 2] a **transportProperties** [Příloha 3]. V prvním souboru se zvolí typ simulace proudění, které je potřeba nastavit na **laminar**, což bude odpovídat laminárnímu modelu simulace. U laminárního modelu musíme brát v potaz jistá zkrácení, jelikož laminární model zcela zanedbává víry, které jsou menší než je element sítě, což znamená, že proudění ovlivní jen víry větší nežli element sítě.

V druhém souboru je potřeba nastavit kinematickou viskozitu tekutiny, což bylo provedeno přesně podle parametrů testovací úlohy [9], kde je uvedena hodnota $0.001 \text{ m}^2\text{s}^{-1}$. Pokud bychom se podívali na viskozitu z inženýrského hlediska a dohledali si zpětně v tabulkách k viskozitě tekutinu, která má reálnou viskozitu nejbližší, tak zjistíme, že v podstatě pracujeme s převodovým olejem SAE 10W o teplotě $-17,8 \text{ }^\circ\text{C}$, jehož tabulková viskozita je $0,001295 \text{ m}^2\text{s}^{-1}$. [6] [1]

Ve vzorci pro Reynoldsovo číslo (3) se vyskytuje parametr charakteristická rychlost

$$|U| = \frac{2*U(0,H/2,t)}{3}, \quad (6)$$

kde H je šířka kanálu a t je čas výpočtu. Vzorec pro výpočet charakteristické rychlosti (6) byl převzat z parametrů testovací úlohy [9]. Po dosazení konkrétních hodnot do vzorce pro charakteristickou rychlost (6)

$$|U| = \frac{2*1,5}{3} = 1m/s$$

a vzorce pro Reynoldsovo číslo (3)

$$Re = \frac{d*|U|}{\nu} = \frac{0,1*1}{0,001} = 100,$$

vyjde hodnota $Re = 100$, což přesně odpovídá hodnotě Re uvedené v práci [9].

Do složky **system** je potřeba vytvořit tři soubory (**controlDict** [Příloha 6], **fvSchemes** [Příloha 4], **fvSolution** [Příloha 5]). Velmi důležité je nastavení souboru **controlDict**, kde se nastavují parametry výpočtu jako je: počáteční čas (**startTime = 0**), konečný čas (**endTime = 6**) a také časový krok simulace δt :

$$\delta t = \frac{Co * \delta x}{|U|} = \frac{1*0,008}{1} = 0,008 s. \quad (7)$$

Kde Co je Courantovo číslo a δx je nejmenší rozměr elementu sítě ve směru proudění (v konkrétním případě $\delta x = 0,008$ m).

Courantovo číslo je důležité pro stabilitu výpočtu, kde musí být splněna podmínka $Co < 1$. Dnes jsou k dispozici řešiče (pimpleFoam), které podporují i $Co > 1$.

Dalším nastavením bude volba proměnného časového kroku (**adjustTimeStep = yes**). Z tohoto důvodu je nutné vhodně nastavit i způsob zápisu výsledků simulace (**writeControl = adjustableRunTime**). Toto nastavení je žádoucí, pokud se data mají zapisovat v přesně definovaný časový interval. Program OpenFoam si automaticky

upraví krok tak, aby se vždy počítal a zapisoval v námi zvoleném zapisovacím intervalu.

Výstupní data budou v komprimované podobě (**writeCompression = compressed**), výstup bude zaznamenáván každých 0.01 sekund (**writeInterval = 0.01**) a maximální Courantovo číslo bude nastaveno na 1 (**maxCo = 1**). Samozřejmě lze nastavit i Courantovo číslo větší než jedna, nicméně by zde mohly vzniknout potíže s výpočtem, ale došlo by k jisté úspoře výpočetního času.

Aby bylo možné získat rychlost, či tlak v bodě, je nutné doplnit soubor **controlDict** o funkci, která bude do nově vytvořeného adresáře **Probes** (sondy) zaznamenávat data rychlosti a tlaku do textového souboru.

Další užitečnou funkcí, kterou je možné doplnit do souboru **controlDict** je funkce **forces**. Pomocí této funkce se budou zaznamenávat síly a momenty působící na obtékané těleso. [1]

Zbylé parametry lze ponechat stejné, jako jsou uvedeny v příloze 6.

Je potřeba se také zmínit o zbylých dvou souborech. Prvním z nich je **fvSolution** [Příloha 5], kde se definují vlastní numerické metody, kterými se budou řídit výpočty fyzikálních veličin (tlak, rychlost). Často se používá řešič **GAMG** (geometricko - algebraický multi grid). V podstatě se jedná o to, že se provede výpočet na hrubé síti, který se poté bere jako počáteční odhad konečného výsledku na jemné síti. Jedná se o metodu prediktor – korektor. Provede se odhad, ten se použije jako výchozí stav a dále se jen porovnává a doladňuje (korektor) přesnost výsledku. Deklarují se zde vlastnosti řešiče, který se stará o řízení celého průběhu výpočtu. Jako řešič bude použit **pimpleFoam** (v souboru pouze **PIMPLE**), který řeší proudění nestlačitelných tekutin a který na základě probíhajících výpočtů upravuje délku kroku. To může vést k úspoře výpočetního času.

Druhým souborem je **fvSchemes** [Příloha 4], v němž se deklarují numerické metody pro diskretizaci diferenciálních operátorů. Pro příklad lze uvést, že pro řešení časových derivací se standardně volí nejjednodušší Eulerova metoda. Dále jsou zde definovány numerické metody pro řešení divergencí a gradientů.

Tyto dva soubory dohromady určují, jak mají být numericky řešeny parciální diferenciální rovnice. Vzhledem k tomu, že se jedná o numerické metody, jejich výsledek pouze konverguje k přesnému řešení. Těmto metodám se také říká iterační a je

nutné definovat jejich ukončení. Nejjednodušším způsobem je zadání chyby (tolerance), kdy se hodnota předcházející a vypočtené veličiny změní právě o danou chybu. Tak je výpočet považován za dostatečně přesný a je ukončen.

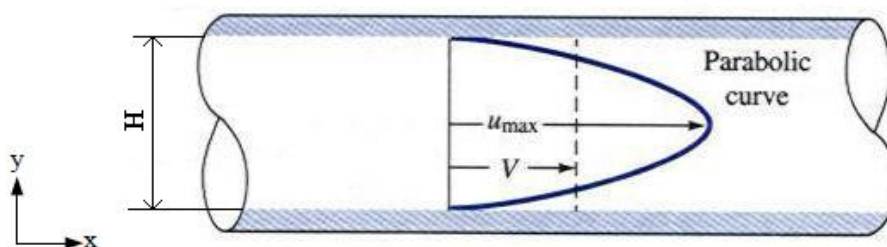
Oba dva soubory fvSchemes a fvSolution je možno použít přesně tak, jak jsou znázorněny v příloze 4 a v příloze 5. [1]

Poslední věcí, kterou je potřeba před započítím samotného výpočtu udělat, je nastavení okrajových (obr. 11) a počátečních podmínek pro rychlost a tlak. Do adresáře s názvem **0** (nultý časový krok) se doplní soubory počátečních a okrajových podmínek tlaku (**p**, Příloha 8) a rychlosti (**U**, Příloha 7). Je možné použít shodné nastavení podmínek pro rychlost jako je v příloze 7 a i pro tlak jako je v příloze 8. I toto nastavení bylo převzato z parametrů testovacích dat [9]. Jediný problém, který bylo potřeba vyřešit, byl parabolický vstupní profil na hranici **Gin**. Tato okrajová podmínka není standardně implementována v programu OpenFoam 1.7.1., tudíž ji bylo potřeba vytvořit nebo sehnat již vytvořenou a implementovat do OpenFoamu.

Na vstupní hranici byla implementována parabolická rovnice pro průběh rychlosti

$$U(0, y, t) = \frac{4 * U_{max} y (H - y)}{H^2}, \quad (8)$$

kde U_{max} je maximální hodnota rychlosti parabolického průběhu, H je šířka kanálu, t je čas a y je souřadnicová osa. [1] [9]



Obr. 12: Parabolický průběh rychlosti. Převzato z [8].

V následujícím přehledu jsou uvedeny okrajové podmínky rychlosti a tlaku na jednotlivých hranicích pro model proudění tekutiny kolem válce v kanálu:

$$\text{Gin:} \quad U(0, y, t) = \frac{4 \cdot 1,5 \cdot y(0,41 - y)}{0,41^2} [\text{ms}^{-1}], \frac{\partial p}{\partial n} = 0 \text{ Pa}$$

$$\text{Gout:} \quad \frac{\partial u}{\partial n} = 0 \text{ ms}^{-1}, p = 0 \text{ Pa}$$

$$\text{Gwall:} \quad u = 0 \text{ ms}^{-1}, \frac{\partial p}{\partial n} = 0 \text{ Pa}$$

$$\text{Gcyl:} \quad u = 0 \text{ ms}^{-1}, \frac{\partial p}{\partial n} = 0 \text{ Pa}$$

V každém souboru, kde definujeme konkrétní hodnoty nějaké fyzikální veličiny, existuje položka *dimensions*, jíž se definuje jednotka fyzikální veličiny. Každé číslo v hranaté závorce reprezentuje mocninu u příslušné SI jednotky. Pomocí hranaté závorky je definována jednotka m^2s^{-1} .

[0 2 -1 0 0 0 0]

Typy okrajových podmínek:

- fixedValue** - Na této oblasti (hranici) se definuje předem definovaná pevná hodnota.
- inletOutlet** - Podmínka inletOutlet předepisuje nulovou veličinu (fixedValue) na místech, kde nám proudění do oblasti vstupuje a zeroGradient, když proudění z oblasti vystupuje.
- slip** - Jestliže je veličina skalár, tak se na hranici předepíše podmínka zeroGradient. Jestliže je veličina vektor, tak se předepíše podmínka fixedValue na její normálovou složku a zeroGradient na složku tečnou.
- zeroGradient** - Předepisuje nulový gradient ve směru normály zkoumané veličiny.
- parabolicVelocity** - Parabolický průběh rychlosti na příslušné hranici.
- Parametry této okrajové podmínky jsou (n – směr proudění, y – směr profilu, maxValue – maximální hodnota příslušné veličiny).

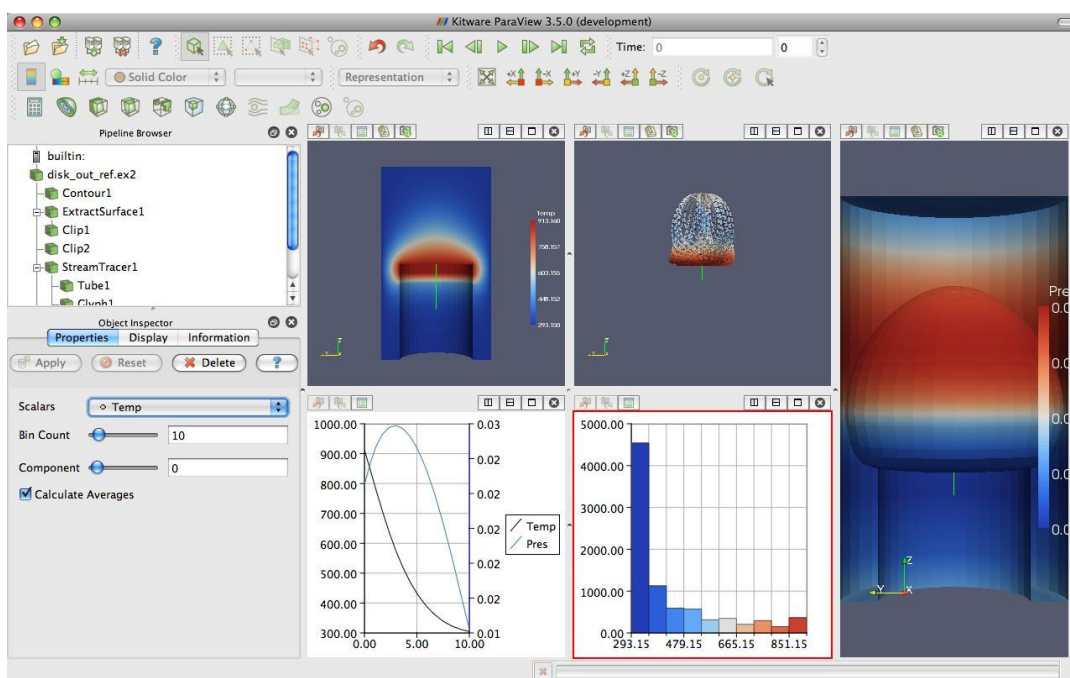
[1]

2.3 Vizualizace výsledků v programu ParaView

V postprocessingu je provedena vizualizace vypočítaných dat a jejich vyhodnocení. Principem vizualizace je reprezentování vypočtených dat pomocí obrazové informace, která je na rozdíl od textových dat více vypovídající, co se týče zhodnocení výsledků a i pro případné pochopení dané problematiky. Vizualizace byla provedena opensource programem ParaView. Tento program umožňuje za použití vhodných filtrů zobrazení detailních a důležitých informací o zkoumaném problému. Může se jednat o zobrazení proudnic, vektorů, isočar, provedení různých řezů objektu, zobrazení průběhu veličin do grafu atp.. Graf může zobrazovat průběh veličiny jako funkci času nebo polohy podél zvolené přímky.

ParaView disponuje čtyřmi základními typy interpretací vypočtených dat. Prvním z nich je **surface** (povrch objektu). Další je **wireframe** (sít' objektu), **Surface with edges** (kombinace zobrazení povrchu a sítě objektu) a posledním typem zobrazení je **volume** (objemové zobrazení objektu – výpočetně nejnáročnější). [2]

Na obr. 13 je zobrazena ukázka vizualizace a případného vyhodnocení pomocí programu ParaView.



Obr. 13: Ukázka možností pro vyhodnocení výsledků vypočtených dat pomocí programu ParaView. Převzato z [2].

ParaView umí vizualizovat také časově proměnná data, na která lze aplikovat velké množství filtrů. Například pomocí jednoho filtru lze získat časově vyprůměrované proudové pole.

Výsledky z programu ParaView lze ukládat jako jednotlivé obrázky, případně jako sérii obrázků, kde jejich spojením může vzniknout video. Další možností je ukládání výsledků do tabulky. [2]

2.4 Ověření výsledků simulace s výsledky testovací úlohy

Pro ověření správnosti nastavení okrajových podmínek a také zda program OpenFoam nenapočítává špatné výsledky, nebo zda je k dispozici dostatečně jemná výpočetní síť, kde nebude docházet k větším odlišnostem mezi výsledky při různé jemnosti výpočetní sítě, se provede ověření našich dat, získaných simulačně s výsledky testovací úlohy, které jsou uvedeny v práci [9]. V práci [9] byl řešen stejný akademický problém, a to právě obtékání válce v uzavřeném kanálu se stejnými parametry, jaké jsem použil v laminárním modelu.

2.4.1 Validace a verifikace CFD modelu

Mezi základní činnosti pro ověření správnosti CFD výpočtu patří tak zvaná validace a verifikace. Základní rozdíl mezi těmito činnostmi je ten, že validace určuje množství chyb během simulace, kdežto verifikace určuje nepřesnosti simulace oproti fyzikální realitě.

Hlavní zdroje chyb CFD simulace tvoří: [11]

- Numerické chyby (zaokrouhlování, ukončení (přesnost) iterativního výpočtu, diskretizační chyby)
- Uživatelské chyby

Hlavní zdroje nepřesností CFD simulace tvoří: [11]

- Nedostatek znalostí o geometrii zkoumaného problému, okrajových podmínkách a proudění tekutin
- Nepřesnosti při získávání dat z fyzického modelu (experimentu)

Při validaci CFD výpočtu nás zajímají tak zvané numerické chyby, které vzniknou při řešení rovnic pomocí numerických metod na počítačích či výpočetních clusterech.

Vzhledem k tomu, že počítače provádějí výpočty s konečným počtem desetinných míst, tak tímto vnáší do výpočtu chybu zaokrouhlování. Další numerickou chybou je konečný počet iterací při numerickém řešení parciálních diferenciálních rovnic. Numerický výpočet se opakuje tak dlouho, dokud bude změna výsledku větší, nežli zadaná tolerance (přesnost). Poslední numerickou chybou je diskretizační chyba, která je způsobena především rozdělením spojitě oblasti na diskrétní. Tato diskrétní oblast se nazývá síť. Pomocí její hrubosti lze usuzovat na přesnost výpočtu. [11]

Při verifikaci CFD výpočtu je potřeba znát základy fyziky proudění tekutin a měli bychom mít k dispozici co nejpřesnější data z měření na reálném objektu, se kterými budeme porovnávat naše vypočtená data. Při řešení reálných, často rozsáhlých a složitých problémů je však obtížné získat vyhovující data z reálných experimentů. Jednou z možností je složitý problém rozdělit na několik jednodušších a na nich jednotlivě provést měření. Při měření na reálném objektu dochází také k jistým nepřesnostem, jelikož žádný měřicí systém nepracuje zcela bezchybně. Tím je zanesena další nepřesnost v porovnávání fyzikálního modelu s numerickým modelem. [11]

Prvním krokem verifikace je ověření základních pravidel proudění tekutin:

- Tekutina proudí od místa s vyšším tlakem k místu s nižším tlakem.
- Tření mezi vrstvami tekutiny způsobí snížení tlaku ve směru proudění
- Rychlost proudění blízko pevné hranice je menší, než dále od ní
- Tlak je větší na zakřivených proudnicích, než uvnitř vírů.
- Proudění se obvykle odtrhává na rozích a ostatních hranách.
- Pokud dochází k odtržení proudění, tak vznikne recirkulační oblast
- Proudění v uzavřeném kanálu s konstantním průřezem se dostane do ustáleného stavu v konečné vzdálenosti.

2.4.2 Vyhodnocení výsledků simulace proudění kolem válce v kanálu

Parametry obtékaného válce v kanálu jsou: viskozita $0,001 \text{ m}^2\text{s}^{-1}$, Reynoldsovo číslo 100, podmínka na vstupní hranici parabolicVelocity a maximální rychlost na vstupu kanálu $1,5 \text{ ms}^{-1}$.

Pro vyhodnocení výsledků bylo použito Strouhalovo číslo (9), odporový součinitel (14), součinitel vztlaku (15) a tlaková difference (10). Těmito parametry

budou porovnány výsledky mého numerického modelu proudění kolem válce v kanálu s výsledky testovací úlohy [9].

Důležitým parametrem pro posouzení výsledků simulace je sledování vlivu počtu elementů sítě na vypočítané řešení. Je-li diskretizační metoda konzistentní, lze očekávat, že s rostoucím počtem elementů také roste přesnost řešení zkoumaného problému. Nicméně simulace na síti s velkým množstvím elementů je výpočetně velmi náročná, tudíž se hledá jejich optimální počet, kde je výsledek simulace již dostatečně přesný a další zvyšování počtu elementů nám přinese navíc akorát výpočetní náročnost, nikoliv výrazné zpřesňování výsledků.

Prakticky se postupuje tak, že se odhadne počet elementů sítě a vyhodnotí se zkoumané parametry. Poté se vytvoří síť s jiným počtem elementů (například dvojnásobným) a opět se vyhodnotí zkoumané parametry. Pokud se výsledky výrazně neliší, vybereme síť s nejmenším počtem elementů a s tou můžeme dále pracovat. Pokud se výsledky výrazně liší, musíme dále zdvojnásobovat počet elementů sítě až do té doby, kdy bude docházet pouze k minimálním změnám zkoumaných parametrů.

Strouhalovo číslo (St) bylo jedním ze zvolených parametrů k vyhodnocení. Toto číslo je bezrozměrné a používá se při popisu periodicky se měnícího proudění. V tekutině musí být zřetelné odtrhávání vírů, aby bylo možné stanovit frekvenci f jejich odtrhávání, která se použije do následujícího vzorce

$$St = \frac{D * f}{|U|}, \quad (9)$$

kde D je charakteristický rozměr (například průměr válce), $|U|$ je charakteristická rychlost proudění, která se vypočítá dle vzorce (6). Po dosazení konkrétních hodnot do vzorce (6):

$$|U| = \frac{2 * 1,5}{3} = 1 \text{ ms}^{-1}.$$

Dalším posuzovaným parametrem bude tlaková difference Δp . Tlakovou diferencii určíme jako rozdíl tlaku před válcem (p_{pred}) a za válcem (p_{za}):

$$\Delta p(t) = p_{pred} - p_{za}, \quad [\text{Pa}]. \quad (10)$$

Tento tlakový rozdíl (10) se určuje v čase

$$t = t_0 + \frac{1}{2f}, \quad [s], \quad (11)$$

kde t_0 je čas, kdy je maximální součinitel vztlaku (15) a f je frekvence odtrhávání vírů. Dalšími parametry jsou odporový součinitel (c_D) a součinitel vztlaku (c_L). Jedná se o bezrozměrné koeficienty, v nichž vystupuje buď odporová síla F_D , nebo vztlaková síla F_L :

$$F_D = \int_S \left(\rho v \frac{\partial \mathbf{u}_t}{\partial n} n_y - p n_x \right) dS, \quad (12)$$

$$F_L = - \int_S \left(\rho v \frac{\partial \mathbf{u}_t}{\partial n} n_x - p n_y \right) dS. \quad (13)$$

Kde S je plocha válce v kanálu, \mathbf{n} normála k ploše S , která se skládá z příspěvků n_x a n_y a tečné rychlosti na tuto plochu \mathbf{u}_t . Z obr. 2 je patrné, že odporová i vztlaková síla se skládají z příspěvku od viskózní a tlakové síly. To je popsáno rovnicemi (12) a (13), kde složka s viskozitou v je příspěvek od viskózní síly a složka s tlakem p je příspěvek od tlakové síly. Odporová síla je potřeba pro výpočet odporového součinitele

$$c_D = \frac{2F_D}{\rho |U|^2 D h} \quad (14)$$

a vztlaková síla je potřeba pro výpočet součinitele vztlaku

$$c_L = \frac{2F_L}{\rho |U|^2 D h}, \quad (15)$$

kde D je průměr válce a h je výška kanálu.

Příklad výpočtu odporového součinitele (14) a součinitele vztlaku (15) je uveden pro čas $\text{Time} = 3,62$. Na obr. 14 je vidět výřez dat ze souboru `forces.dat`, kde jsou hodnoty tlakové a viskózní síly rozloženy jako příspěvky do jednotlivých souřadných systémů.

```
# Time      forces(pressure, viscous) moment(pressure, viscous)
:           :
3.62      (((0.0100671 0.00062677 -1.25254e-25) (0.00305475 -0.000129451 -3.48355e-22)) ...
3.63      (((0.0100634 0.00115778 -1.47067e-25) (0.00305306 -5.36162e-05 -3.49808e-22)) ...
3.64      (((0.0100792 0.00163957 -1.70547e-25) (0.00304886 2.79373e-05 -3.51174e-22)) ...
3.65      (((0.010095 0.0020567 -1.92914e-25) (0.00305023 0.000105728 -3.52553e-22)) ...
3.66      (((0.0101215 0.00239328 -2.10827e-25) (0.00305288 0.00017941 -3.54041e-22)) ...
3.67      (((0.010158 0.0026371 -2.2556e-25) (0.00305669 0.000247317 -3.55196e-22)) ...
3.68      (((0.010196 0.00277086 -2.32455e-25) (0.00306033 0.000307534 -3.56254e-22)) ...
:           :
```

Obr. 14: Struktura souboru forces.dat

Síla F_D (12) bude vypočítána jako

$$F_D = F_{Dtlak} + F_{Dvisk} = 0,0100671 + 0,00305475 = \mathbf{0,01312185\ N},$$

kde F_{Dtlak} je příspěvek tlakové síly a F_{Dvisk} je příspěvek viskózní síly do souřadného směru X (obr. 2). Síla F_L (13) bude vypočítána jako

$$F_L = F_{Ltlak} + F_{Lvisk} = 0,00062677 + (-0,000129451) = \mathbf{0,000497319\ N},$$

kde F_{Ltlak} je příspěvek tlakové síly a F_{Lvisk} je příspěvek viskózní síly do souřadného směru Y (obr. 2).

Výpočet odporového součinitele c_D (14) po dosazení ($h = 0,08\ \text{m}$, $D = 0,1\ \text{m}$, $\rho = 1\ \text{kg/m}^3$, $|U| = 1\ \text{m/s}$ (6)) je

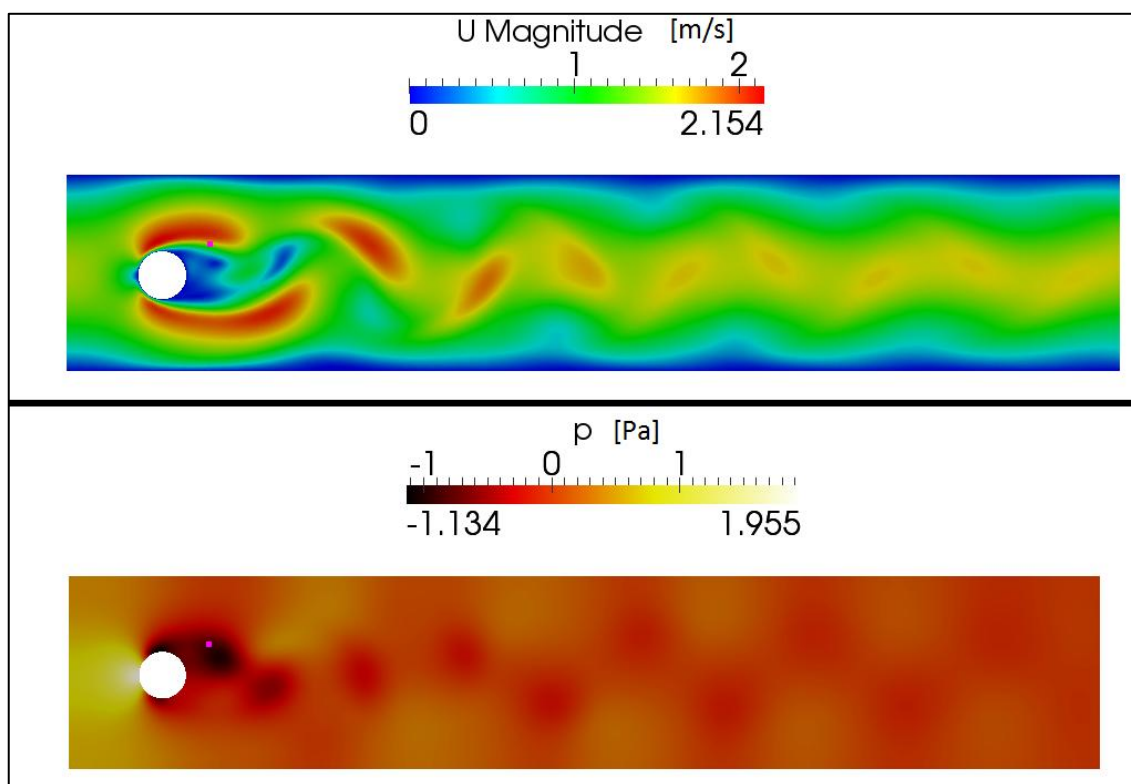
$$c_D = \frac{2F_D}{\rho|U|^2 D h} = \frac{2 \cdot 0,01312185}{1 \cdot 1^2 \cdot 0,1 \cdot 0,08} = \mathbf{3,280463}$$

a výpočet součinitele vztlaku c_L (15) je

$$c_L = \frac{2F_L}{\rho|U|^2 D h} = \frac{2 \cdot 0,000497319}{1 \cdot 1^2 \cdot 0,1 \cdot 0,08} = \mathbf{0,12433}.$$

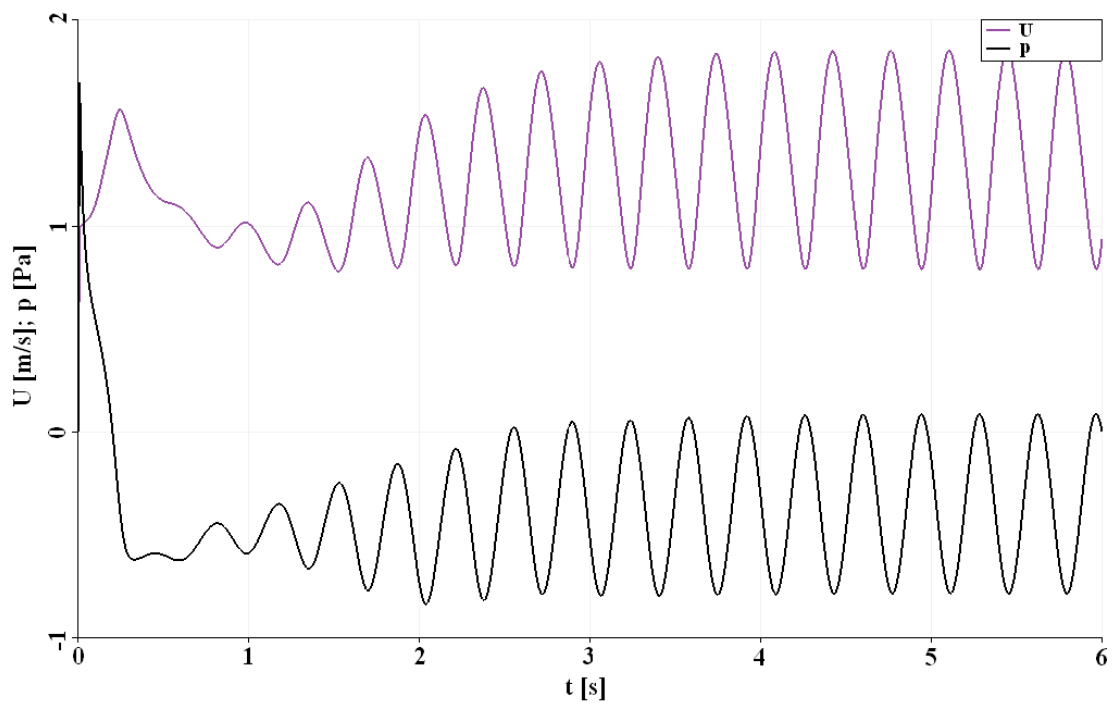
Příklad výpočtu Strouhalova čísla pro jednu konkrétní jemnost sítě, která má 95894 elementů, bude uveden níže. Pomocí programu ParaView bude provedena vizualizace na základě které budou stanoveny potřebné veličiny pro výpočet Strouhalova čísla. V tomto případě se bude jednat především o frekvenci.

Zvolený bod pro určení frekvence odtrhávání vírů (čtvereček růžové barvy) je na obr. 15.

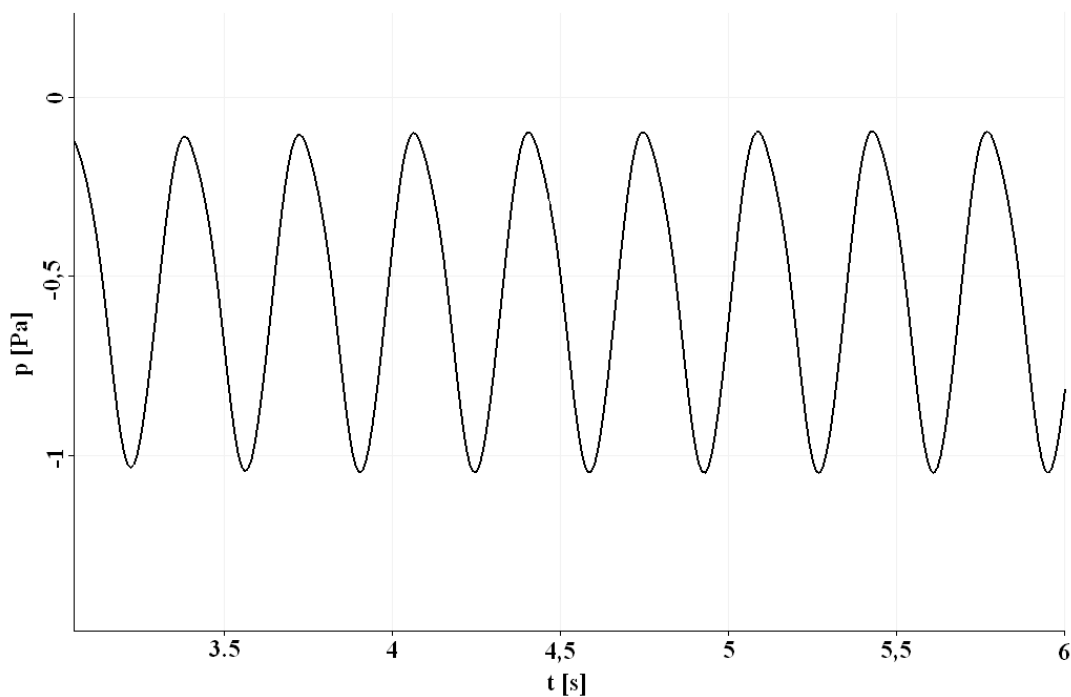


Obr. 15: Bod pro učení frekvence odtrhávání vírů

V grafu (obr. 16) je zobrazen průběh tlaku a rychlosti v celém časovém rozsahu simulace. Lze pozorovat, že od určitého času (v mém případě 3,5 s) je průběh rychlosti, či tlaku periodický a více méně harmonický. Časové průběhy se v programu ParaView zobrazí pomocí funkce *PlotSelectionOverTime*.



Obr. 16: Průběh rychlosti a tlaku v celém času simulace. Zvolený bod pro vykreslení těchto průběhů je na obr. 15.



Obr. 17: Detail průběhu tlaku v bodě zobrazeném na obr. 15.

Perioda odtrhávání vírů (T) se stanoví z průběhu tlaku v ustáleném stavu za použití vzorce:

$$T = \frac{t_2 - t_1}{N} = \frac{5,95 - 3,22}{8} = \mathbf{0,34125\ s}, \quad (16)$$

kde t_1 je čas prvního bodu periody (v mém případě jsem volil vrcholy minima z průběhu tlaku), t_2 je čas posledního bodu periody a N je počet period.

Z periody se jednoduše stanoví frekvence odtrhávání vírů

$$f = \frac{1}{T} = \frac{1}{0,34125} = \mathbf{2,9304\ Hz}. \quad (17)$$

Nyní lze vypočítat Strouhalovo číslo dosazením do vzorce (9):

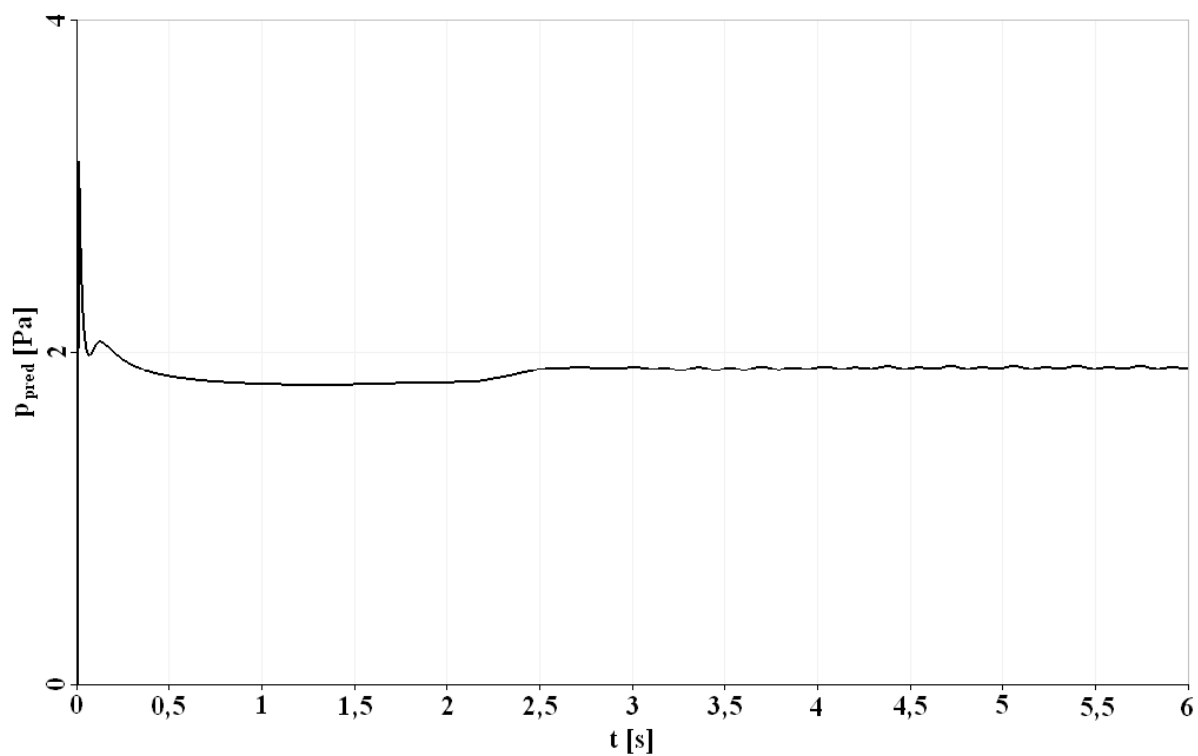
$$St = \frac{D \cdot f}{|U|} = \frac{0,1 \cdot 2,9304}{1} = \mathbf{0,29304}.$$

Tlaková difference (10) bude stanovena v bodech 1 a 2 zobrazených na obr. 18.



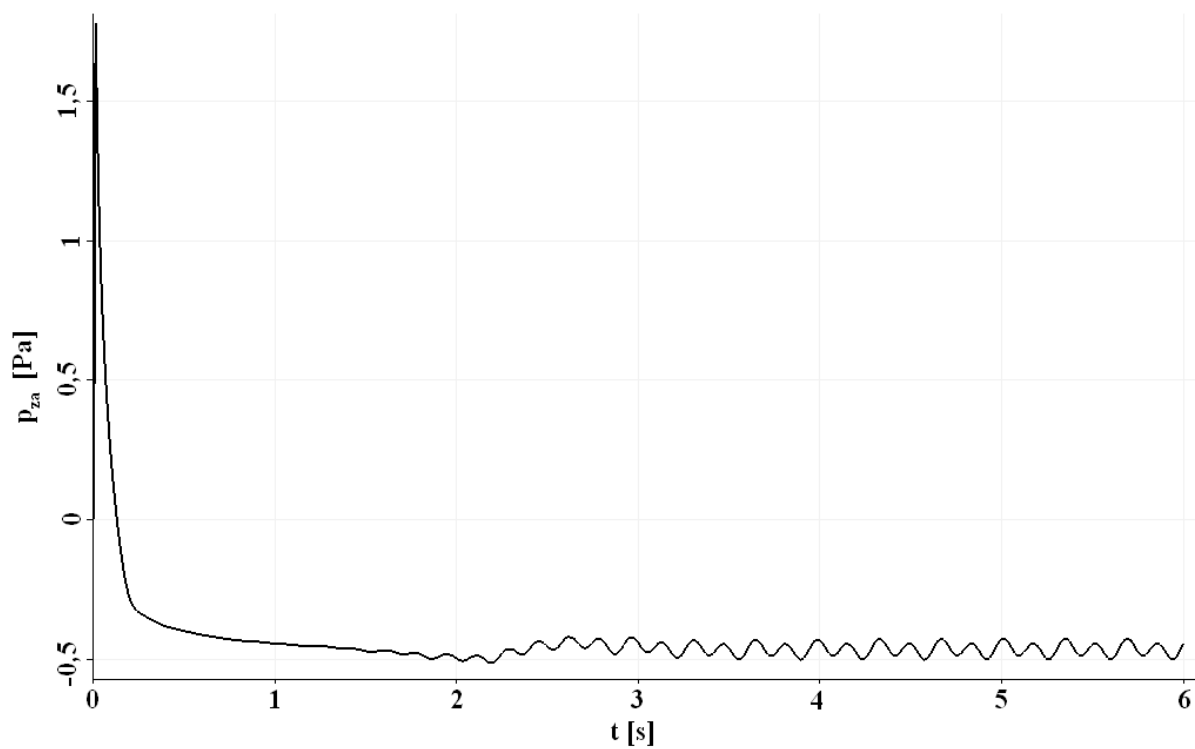
Obr. 18: Body pro určení tlakové difference

V grafu (obr. 19) je zobrazen průběh tlaku (p_{pred}) v bodě 1 (obr. 18).



Obr. 19: Průběh tlaku před válcem

a v grafu (obr. 20) je zobrazen průběh tlaku (p_{za}) v bodě 2 (obr. 18).



Obr. 20: Průběh tlaku za válcem

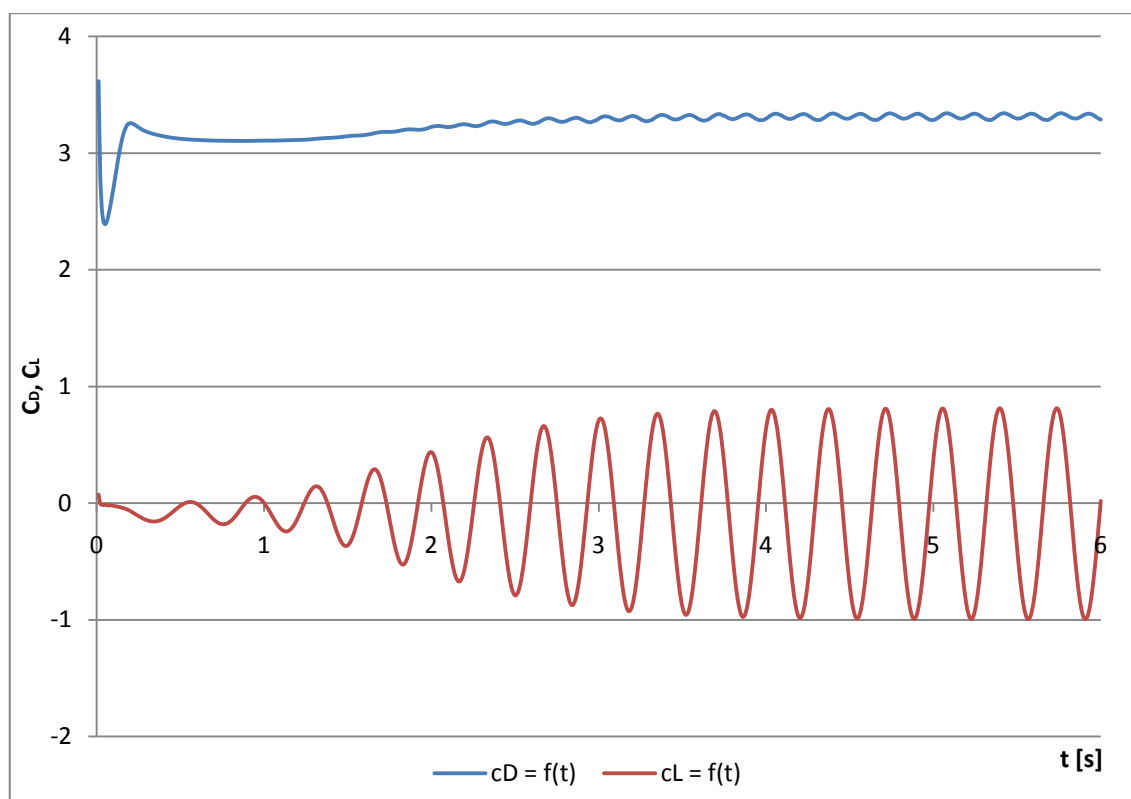
Z grafu (obr. 21) bude stanoven čas t_0 ($t_0 = 5,74$ s), kdy je maximální součinitel vztlaku (c_{Lmax}). Frekvence odtrhávání vírů f je 2,9304 Hz (17). Nyní lze vypočítat čas t pro stanovení tlakové difference:

$$t = t_0 + \frac{1}{2f} = 5,74 + \frac{1}{2 \cdot 2,9304} = 5,74 + 0,17 = \mathbf{5,91\ s.}$$

V čase $t = 5,91$ sekund se z grafu na obr. 19 odečte hodnota tlaku před válcem ($p_{pred} = 1,90641$ Pa) a z grafu na obr. 20 se odečte hodnota tlaku za válcem ($p_{za} = -0,47274$ Pa). Po dosazení do vzorce pro tlakovou diferenci (10):

$$\Delta p(t) = p_{pred} - p_{za} = 1,90641 + 0,47274 = \mathbf{2,398\ Pa}$$

Z grafu (obr. 21) je vidět, že rozkmit součinitele vztlaku je mnohem větší, než rozkmit odporového součinitele. Součinitel vztlaku dosahuje kladných i záporných hodnot na rozdíl od druhého odporového součinitele, který se pohybuje pouze v kladných hodnotách.



Obr. 21: Průběh odporového součinitele c_D a součinitele vztlaku c_L jako funkce času

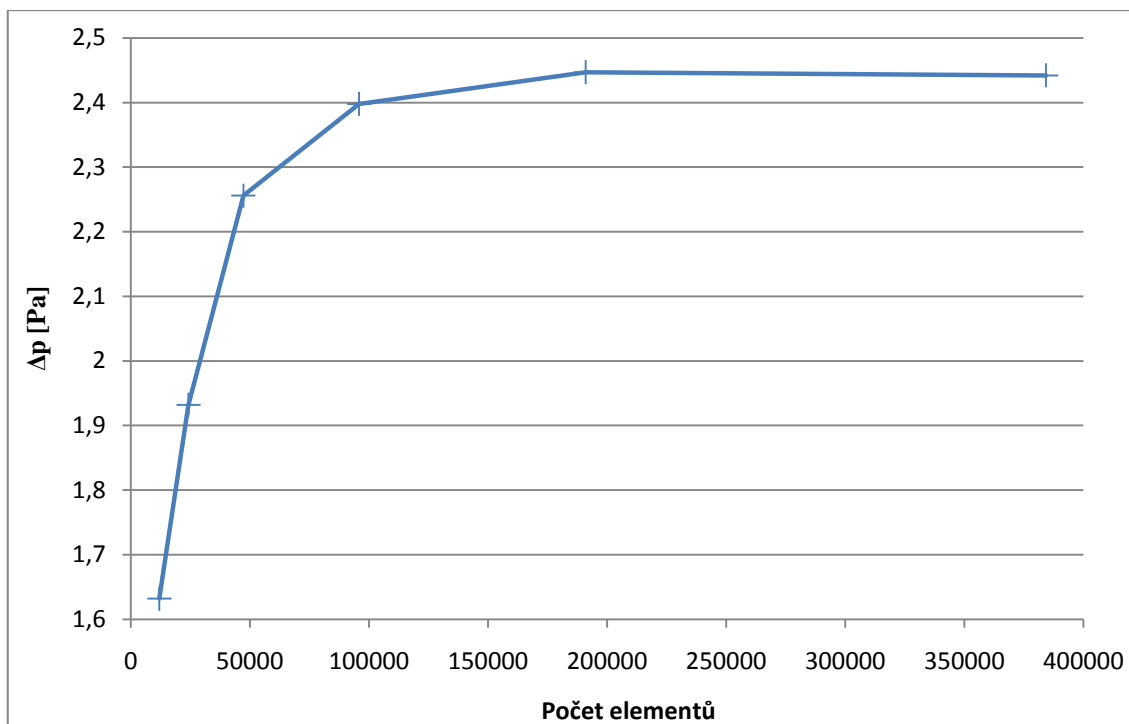
Stejně tak, jak bylo stanoveno Strouhalovo číslo a tlaková diference pro tuto konkrétní síť (95894 elementů), se bude pokračovat i v případech s hrubší i jemnější sítí a vypočtené výsledky budou zaznamenány do tabulky (tab. 2). Červeně vyznačené pole v tabulce označují hrubosti sítě, kde již nedochází k přílišným odlišnostem mezi parametry. Nejlépe to je vidět na tlakové diferenci Δp (obr. 22).

Počet elementů	Výp. čas [h:mm]	fodtrhávání [Hz]	St	C _{Dmax}	C _{Lmax}	Δp [Pa]
11980	0:05	XX	XX	2,7336	-0,3746	1,632
24228	0:13	2,9325	0,29325	2,8123	0,7428	1,932
47256	0:31	2,9508	0,29508	3,2053	0,6623	2,256
95894	1:18	2,9304	0,29304	3,3426	0,814	2,398
190970	4:24	2,9498	0,29498	3,3311	0,9684	2,447
384310	13:35	2,9851	0,29851	3,2615	0,9386	2,442

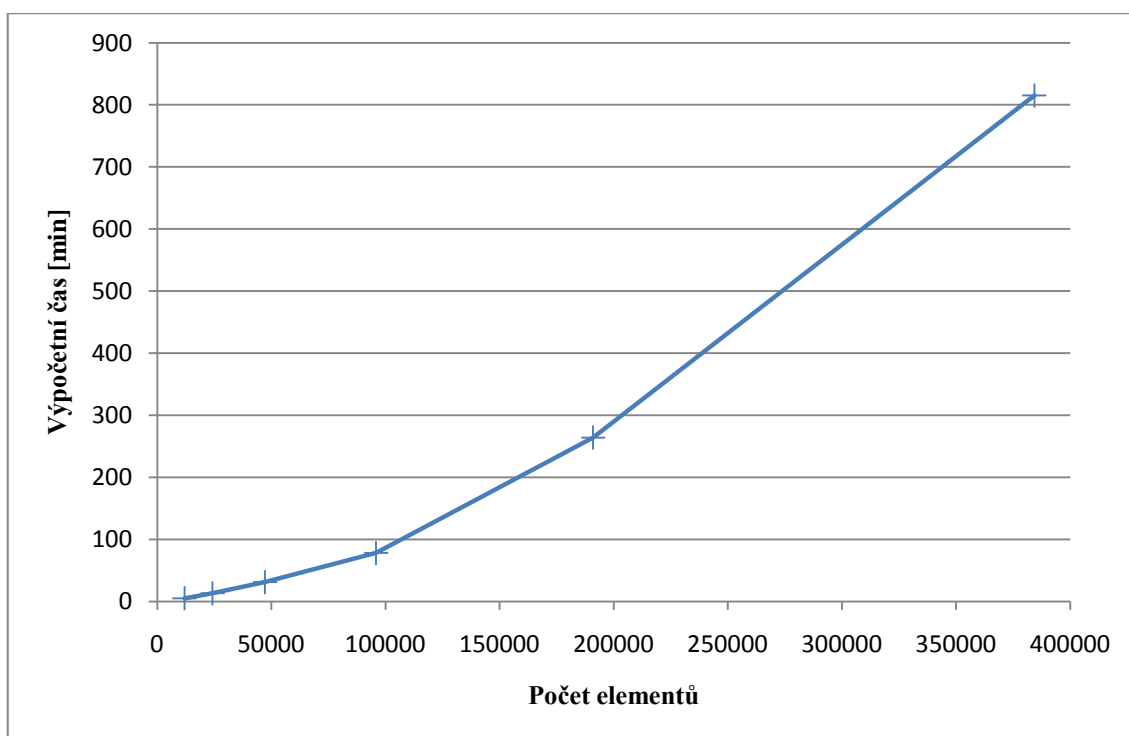
Tab. 2: Tabulka s výsledky simulace

Simulace byla prováděna na notebooku s procesorem Intel Core 2 (T5500 @ 1.66 GHz) a 1GB RAM. Jako operační systém je nainstalována distribuce Linuxu - Ubuntu 10.04.

Dle grafu pro diferenci tlaku (obr. 22) jako funkce počtu elementů sítě je vidět, že od sítě s počtem elementů 190970 již nedochází k výrazným změnám tlakové diference, tudíž síť s touto hrubostí je vhodné používat pro další možné simulační výpočty. Při použití sítě s větším počtem elementů se výsledek bude pouze minimálně zpřesňovat za cenu zvyšování výpočetní náročnosti (obr. 23).



Obr. 22: Závislost tlakové difference na počtu elementů sítě



Obr. 23: Závislost výpočetní náročnosti na počtu elementů sítě

Z grafu na obr. 22 byla stanovena hranice množství elementů sítě (190970), kde se dají považovat výsledky numerické simulace proudění za dostatečně přesné, tudíž je lze použít pro porovnání s výsledky testovací úlohy [9] uvedenými v tab. 3.

V tabulce (tab. 3) jsou zahrnuty výsledky simulací různých vědeckých skupin s různými metodami přístupu a různými hrubostmi sítě.

Graf na obr. 23 ukazuje závislost výpočetní náročnosti na počtu elementů a jasně říká, že čím je více elementů sítě, tím je větší výpočetní náročnost. Z tohoto důvodu se hledá nejmenší možný počet elementů sítě, kde jsou výsledky dostatečně přesné.

	Unknowns		c_{Dmax}	c_{Lmax}	St	ΔP	Mem.	CPU time	MFlop rate
	Space	Time							
1	267476	67	3.2224	0.9672	0.2995	2.4814	—	—	1600 PEAK
	267476	34	3.2030	0.9223	0.2941	2.4664	—	—	
	267476	18	3.1605	0.8026	0.2901	2.4466	—	—	
	68212	67	3.2171	0.9591	0.2995	2.5009	—	—	
	17732	68	3.2168	0.9295	0.2979	2.5573	—	—	
3	12800	34	3.2200	0.9720	0.2960	2.4700	2.5	789	75 PEAK
4	297472	670	3.2460	0.9840	0.2985	2.4900	137	6600	445 LINP
	297472	338	3.2710	0.9800	0.2959	2.4870	137	3400	
	297472	172	3.3200	0.9720	0.2907	2.4810	137	1700	
	75008	670	3.2410	0.9910	0.2985	2.5020	73	2350	
	19008	674	3.2320	1.0260	0.2967	2.5320	57	1350	
6	332640	12000	4.1210	1.6120	0.3330	3.1420	10	10086	13 LINP
	85140	6000	4.7330	2.0600	0.3380	3.4300	2.6	1259	
7a	294912	36	3.2358	1.0069	0.3003	2.4892	75	6167	13 LINP
	294912	19	3.2356	1.0000	0.2973	2.4871	75	6391	
	294912	10	3.2152	0.9028	0.2881	2.4715	75	4994	
	73728	36	3.2443	1.0261	0.2994	2.4929	19	1946	
	18432	36	3.2706	1.0695	0.2968	2.5035	5	445	
8a	29084	66	3.2240	1.0060	0.3020	2.4860	11	4992	8.3 LINP
	29084	33	3.2470	1.0740	0.3030	2.5010	11	3777	
	29084	16	3.2900	1.2500	0.3130	2.5700	11	3217	
	8764	66	3.1740	0.9640	0.3000	2.4630	3.6	1000	
	2978	70	2.8920	0.5540	0.2890	2.2870	1.5	339	
9a	240000	5000	3.2267	0.9862	0.3017	2.4833	53	32500	34 LINP
	60000	10000	3.2232	0.9830	0.3012	2.4773	10	8550	
	60000	5000	3.2232	0.9832	0.3012	2.4773	10	4500	
	60000	2500	3.2232	0.9836	0.3012	2.4773	10	3400	
	15000	5000	3.2058	0.9651	0.2994	2.4587	2.5	3240	
10	667264	612	3.2314	0.9999	0.2973	2.4707	128	8545	90 LINP
	667264	204	3.2351	1.0123	0.2957	2.4734	128	2850	
	667264	68	3.2771	1.1205	0.2997	2.4961	128	1065	
	167232	188	3.2498	1.0081	0.2927	2.4410	32	655	
	42016	164	3.2970	0.8492	0.2713	2.3423	8	147	
13b	25410	6755	3.1822	1.0692	0.2960	2.6066	5.1	44710	90 LINP
	25410	3877	3.1895	1.0883	0.2968	2.6057	4.8	27175	
	25410	1678	3.2043	1.1268	0.2979	2.5307	4.7		
	12738	6799	3.1945	1.1233	0.2941	2.6140	2.9	13045	
	6562	7223	3.1317	1.2961	0.2768	3.0253	1.8		
15a	432960	7790	3.0804	0.7256	0.2778	2.1330	4.4	108844	7.4 LINP
	108240	4003	3.1677	0.6880	0.2646	2.0954	1.1	34876	
	108240	3859	3.1096	0.8249	0.2841	2.1105	1.1	58003	
	27060	1985	3.2544	0.5658	0.2336	1.9727	0.3	3796	
	27060	1670	3.1759	0.7656	0.2740	1.9961	0.3	4188	
lower bound			3.2200	0.9900	0.2950	2.4600			
upper bound			3.2400	1.0100	0.3050	2.5000			

Tab. 3: Tabulka s výsledky numerických simulací testovací úlohy. Převzato z [9].

Ze shrnutí výsledků v tab. 4 je vidět, že výsledky diplomové práce se shodují s výsledky testovací úlohy [9] s odchylkou do 6 %. Tím je úspěšně verifikováno použití programu OpenFoam pro numerické modelování proudění nestlačitelné tekutiny kolem válce v uzavřeném kanálu.

	St	C _{Dmax}	C _{Lmax}	Δp [Pa]
Diplomová práce	0,29851	3,2615	0,9386	2,442
Testovací úloha [9] (Tab. 3)	0,29500- 0,30500	3,2200 - 3,2400	0,9900 - 1,0100	2,460 - 2,500
Testovací úloha [9] experiment	0,284 – 0,29	X	X	X

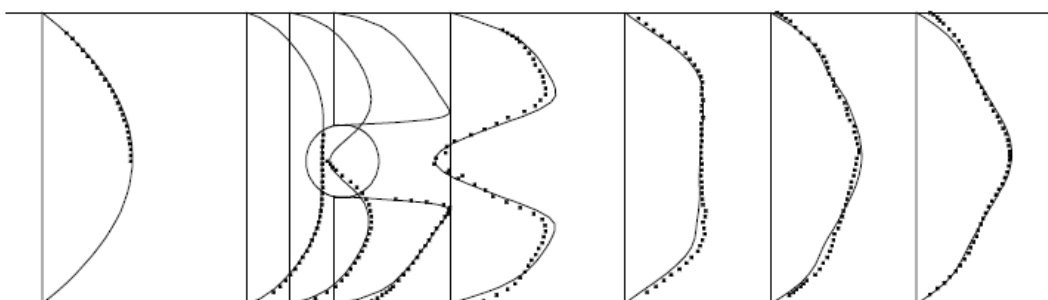
Tab. 4: Porovnání výsledků numerické simulace proudění v diplomové práci s výsledky testovací úlohy [9] v tab. 3 a experimentálně získanými daty [9].

Další zpřesnění výsledků by mohlo nastat, kdyby byl nastaven delší simulační čas. Z času pro určení tlakové difference $t = 5,91$ sekund je vidět, že vychází až ke konci celkového simulačního času (6 sekund), tudíž lze předpokládat, že po prodloužení simulačního času by mohlo dojít ke zpřesnění výsledků. K dalšímu zpřesnění výsledků by mohlo dojít při dalším zjemňování sítě.

2.4.3 Porovnání rychlostních profilů

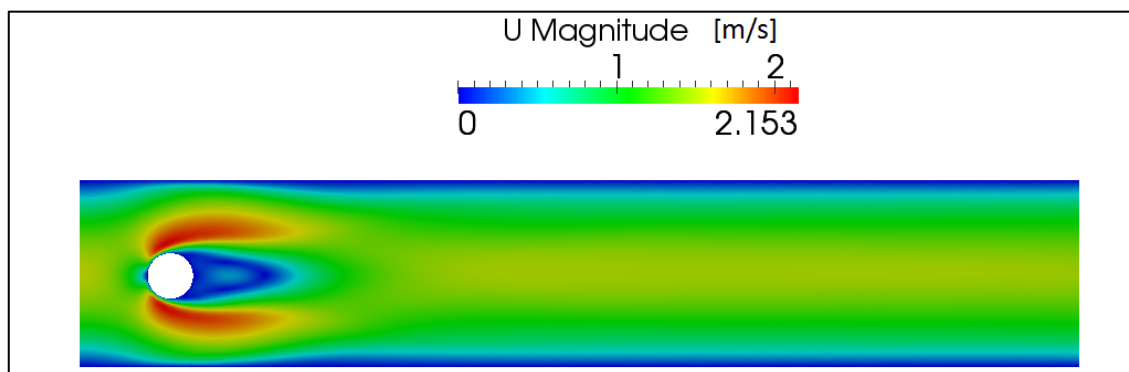
Další možností, jak validovat výsledky naší simulace je porovnávání rychlostních profilů v konkrétních místech na simulovaném problému s výsledky získanými experimentálně. Tyto experimentálně získané výsledky jsou zaznamenány v práci [9].

Jedná se o průběhy průměrné rychlosti získané během celé doby simulace podél zvolených přímk. Data získaná experimentálně a místa jejich měření jsou zaznamenána na (obr. 24).

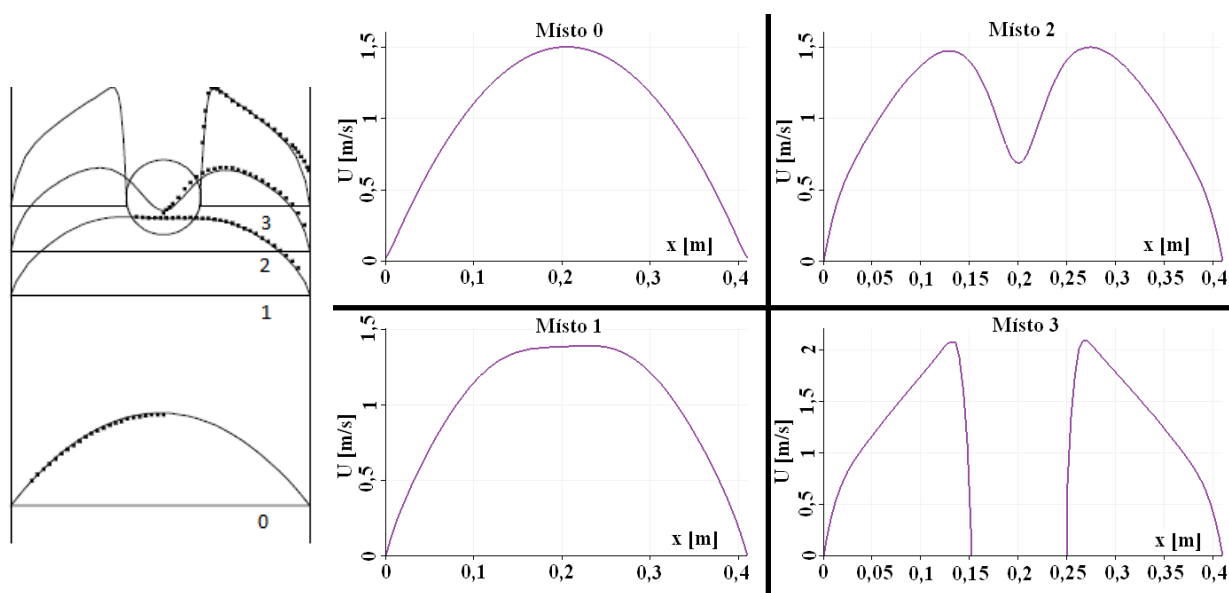


Obr. 24: Rychlostní profily testovací úlohy [9] podél příslušných přímek.

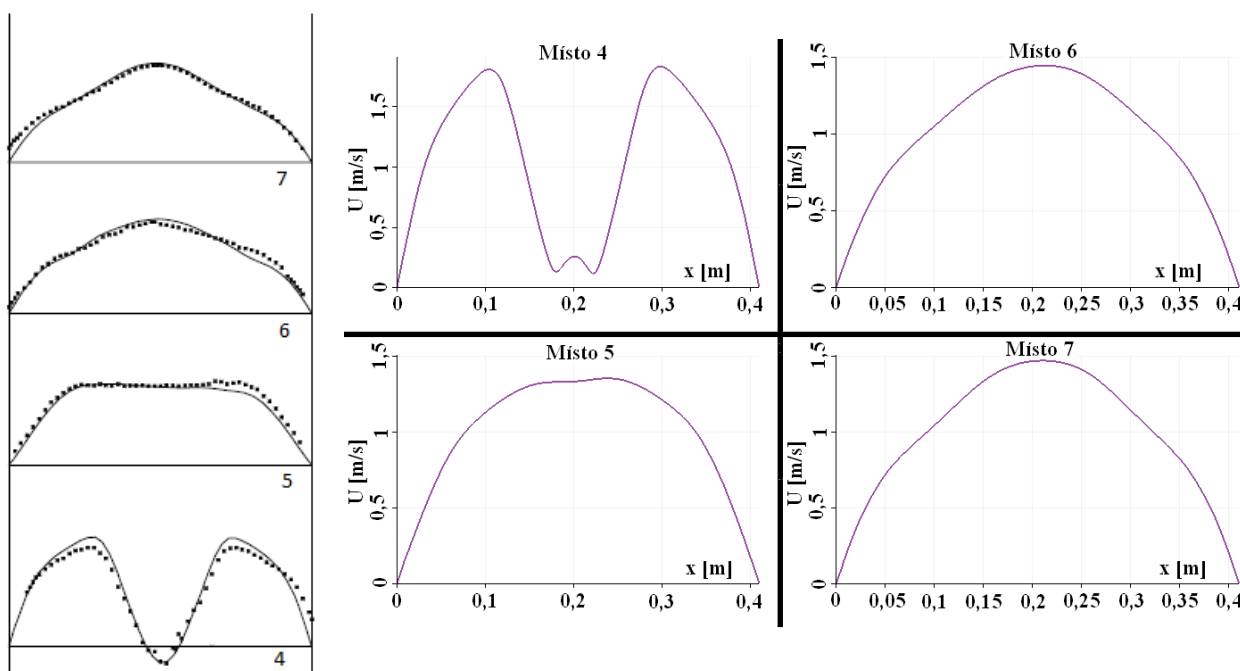
Vizualizace tohoto problému bude řešena pomocí programu ParaView. Filtr pro výpočet vyprůměrovaného rychlostního pole v celém časovém rozsahu simulace se nazývá *TemporalStatistic*. Rychlostní profil podél zvolené přímky se získá pomocí filtru *PlotOverLine*. Na následujících obrázcích budou zobrazeny mé nasimulované rychlostní profily. Jednotlivá místa měření rychlostních profilů byla označena zleva 0 .. 7.



Obr. 25: Vyprůměrované rychlostní pole



Obr. 26: Porovnání graficky zobrazených rychlostních průběhů v místech 0 – 3 (vlevo – výsledky testovací úlohy [9], vpravo výsledky diplomové práce)



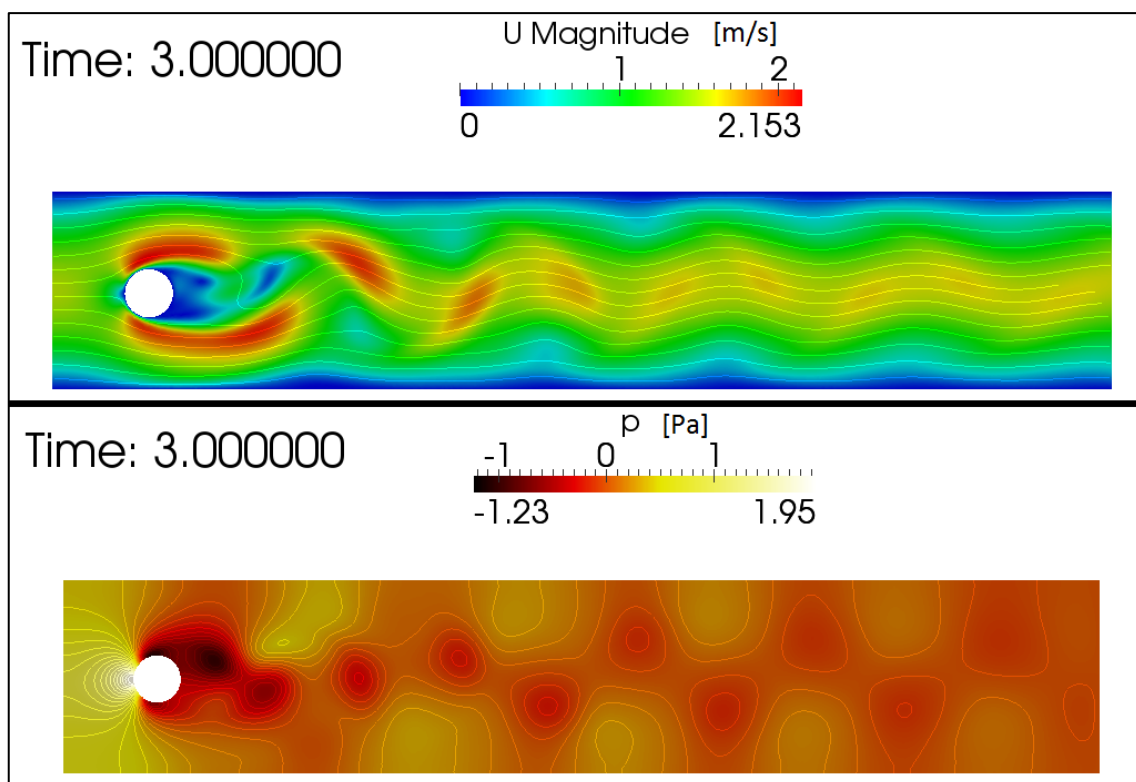
Obr. 27: Porovnání graficky zobrazených rychlostních průběhů v místech 4 – 7 (vlevo – výsledky testovací úlohy [9], vpravo výsledky diplomové práce)

Porovnáním rychlostních profilů testovací úlohy z práce [9], které jsou výřezově umístěny vždy v levé části obr. 26 a obr. 27, s rychlostními profily získanými z mé simulace, umístěnými v pravé části, lze konstatovat, že se mezi sebou nijak výrazně neliší. To představuje další argument pro to, že zvolený výpočetní software OpenFoam a veškerá nastavení důležitých parametrů výpočtu, včetně okrajových podmínek, byly provedeny korektně.

2.5 Vizualizace proudění kolem statického válce v kanálu

Výsledky simulace proudění laminárního modelu kolem válce v kanálu jsou zobrazeny na obr. 28. Konkrétně se jedná o čas simulace $t = 3$ sekundy. Celá simulace byla puštěna na síti, která měla 190 970 elementů. Proč byla zvolena pro vizualizaci právě tato jemnost sítě je podrobně rozebráno v kapitole 2.4.

Na obr. 28 je vidět, že se za válcem začínají odtrhávat víry, které proudění ovlivňují. Rychlost je zvýrazněna vektory rychlosti (filtr *Glyph*) a tlak je zvýrazněn proudnicemi (filtr *stream tracer*). Místa, kde se tvoří víry, jsou poznat podle toho, že uprostřed víru je nulová rychlost, což odpovídá modré barvě rychlosti. Maximální rychlost proudění v kanálu dosahuje 2.16 ms^{-1} .



Obr. 28: Vizualizace proudění kolem válce v kanálu v čase $t = 3$ s. Vrchní obrázek zobrazuje rychlost a spodní tlak proudění.

3 Obtékání kmitající překážky

Tato část bude věnována numerické simulaci obtékání kolem kmitající překážky. Problematika, kde dochází k pohybu nějaké části systému, je z praktického hlediska velmi častá. Příkladem může být numerické modelování kmitání věží, mostů, případně mostních lan. Z tohoto důvodu je užitečné, aby zde byl nasimulován alespoň jednoduchý problém, pomocí kterého si lze vyzkoušet příslušnou problematiku a odhalit tak případné nástrahy, které se mohou v tomto případě vyskytnout.

3.1 Dynamická síť

Existují dva přístupy pro řešení dynamické sítě a to, zda se bude řešit případ, kdy je uvažována síť s pevnou topologií, kde žádné elementy nevznikají ani nezanikají, nebo se bude řešit případ s proměnnou topologií, kde se z různých důvodů musí elementy sítě přidávat nebo ubírat. V tab. 5 jsou uvedeny možnosti pro přístup a práci se sítí, které program OpenFoam nabízí.

Práce se sítí	
Pevná topologie	Proměnná topologie
staticFvMesh	linearValveFvMesh
dynamicMotionSolverFvMesh	linearValveLayersFvMesh
dynamicInkJetFvMesh	mixerFvMesh
----	movingConeTopoFvMesh

Tab. 5: Možnosti práce se sítí v programu OpenFoam [13]

3.1.1 Pevná topologie

Prvním přístupem pro práci s pevnou topologií sítě je knihovna **staticFvMesh**, kde nedochází k žádným dynamickým změnám. Jedná se o případ statické sítě. Dalším přístupem je **dynamicMotionSolverFvMesh**, kde je řešen pohyb sítě za pomoci zvoleného matematického modelu a modelu difuzivity. Používá se pro menší deformace sítě. [13]

Pro matematické řešení pohybu sítě budou zmíněny dva přístupy:

- Pružinová analogie, kde jsou elementy sítě spojeny fiktivními pružinami.
- Laplaceova analogie, kde se pohlíží na oblast jako na hmotný celek, kde hmota nikde nevzniká ani se nikam neztrácí a pouze se přesouvá

vlivem deformace. V OpenFoamu je tento přístup označen jako řešič **velocityLaplacian** [16]

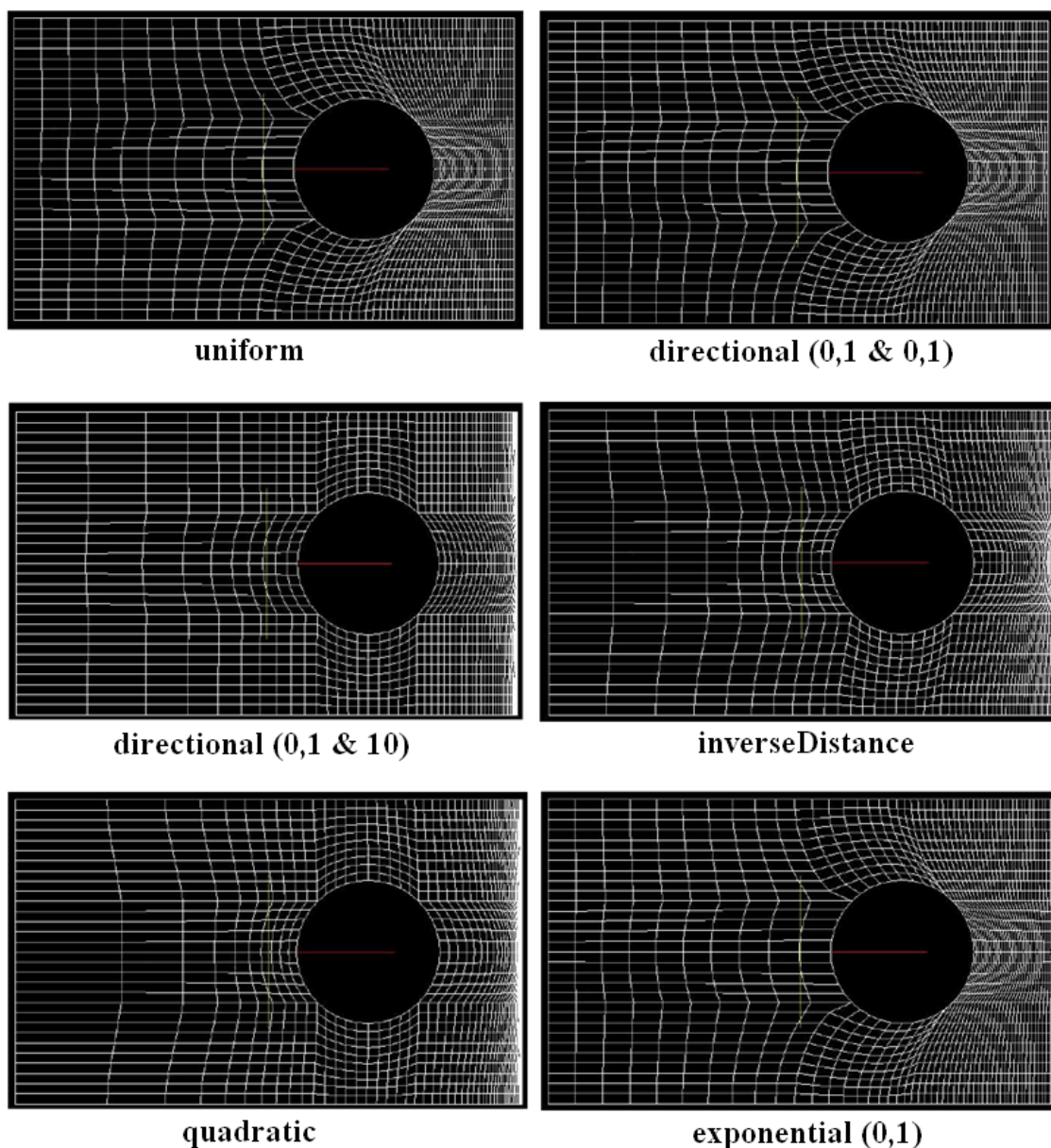
$$\nabla \cdot (\gamma \nabla \mathbf{u}_s) = 0, \quad (18)$$

kde \mathbf{u}_s je vektor rychlosti pohybu sítě a γ je parametr difuzivity.

V obou výše zmíněných přístupech hraje roli parametr s názvem difuzivita (18), pomocí které se určí, jak se bude deformovaná síť chovat. V programu OpenFoam jsou implementovány následující modely difuzivity. [13]

- **uniform** – Zde jsou všechny elementy sítě deformovány se stejným koeficientem, nicméně záleží, jak jsme daleko od pohybující se hranice.
- **directional** – V tomto případě se nejvíce deformují elementy ve směru pohybu válce. Jakým způsobem se elementy deformují, předepisují příslušné koeficienty.
- **inverseDistance** – K největší deformaci elementů sítě dochází na protější hranici od pohybujícího se tělesa.
- **linear** – Deformace je šířena lineárně s tím, že k největší deformaci dochází naproti pohybující se hranici.
- **quadratic** – Podobný případ jako difuzivita linear jen se deformace šíří kvadraticky.
- **exponential** – Poslední uvedený případ je opět podobný modelům difuzivity linear a quadratic, jenže zde se deformace šíří exponenciálně.

Porovnání deformací pohyblivé sítě při různých nastaveních difuzivity je zobrazeno na obr. 29.



Obr. 29: Příklady chování pohyblivé sítě při použití různé difuzivity. Převzato z [13].

Posledním přístupem pro práci s pevnou topologií sítě je knihovna **dynamicInkJetFvMesh**, kde vystupují jako parametry difuzivita a matematický model pohybu sítě. Je zde definován harmonický pohyb podél zvolené referenční plochy. [13]

3.1.2 Proměnná topologie

V odstavci 3.1.1 byla rozebrána problematika práce s pevnou topologií sítě, která má omezení na rozsah pohybu pohybující se hranice. Při větších amplitudách pohybu je nutné použít knihovny, které umožňují měnit topologii sítě. První knihovnou

je **linearValveFvMesh**, která je určena pro případy podobné numerickému modelování pohybu pístu ve válci, kde je přítomna statická hranice (např. blok motoru) a lineárně se pohybující hranice (např. píst). Tato knihovna zajistí korektní smýkání hranic, při kterém dochází v každém časovém kroku k rozpojení a spojení hranice mezi pevnou a pohyblivou hranicí. Další knihovnou pro řešení pohybu proměnné topologie sítě je **linearValveLayersFvMesh**, která navíc oproti knihovně **linearValveFvMesh** dovoluje přidávání nebo odebírání vrstev pod, případně nad pohybující se hranicí (pístem). Další knihovnou je **mixerFvMesh**, která je použitelná v případech, kde dochází ke smýkání hranic vlivem rotačního pohybu. V tomto případě existuje rotační pohyblivá hranice (rotor) a statická hranice (stator). Poslední knihovnou pro práci s proměnnou topologií pohybující se sítě je **movingConeTopoFvMesh**, kde je pohyb hranice definován sinusově. Jsou zde definovány regiony, kdy při zvolené tloušťce elementů sítě (deformaci sítě) dochází k jejich odebírání, případně přidávání. [13]

3.2 Kmitající válec v kanálu

V kapitole 2 byl rozebrán numerický model pro obtékání statického válce v kanálu. Nicméně o něco zajímavější případ je, pokud se válec v kanálu pohybuje. Tento pohyb bude zvolen jako kmitavý s amplitudou A a frekvencí f_k :

$$y(t) = A * \sin(\omega t), \quad (19)$$

kde $\omega = 2\pi f_k$ je úhlová rychlost.

3.2.1 Nastavení numerické simulace

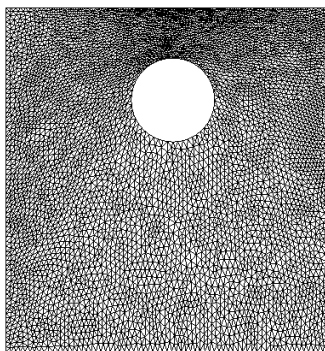
Generování sítě, ať už se jedná o geometrii řešeného modelu (obr. 9), použitý software (Gmsh), či další problematiku s tímto spjatou, je shodný, jako v kapitole 2.1. Mnoho poznatků bude použito dále naprosto beze změn.

Pro nastavení parametrů vlastního výpočtu bylo použito mnoho shodného s kapitolou 2.2.

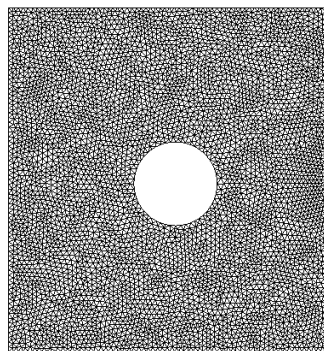
Dynamická síť, jak už název napovídá, se skládá z jedné nebo i více pohyblivých částí, jejichž pohyb je nutné definovat. K tomu slouží také okrajové podmínky, které většinou nejsou v základu OpenFoamu obsaženy, tudíž je nezbytné tam tyto podmínky doplnit.

V případě dynamické sítě je potřeba do adresářové struktury OpenFoamu přidat dva soubory a některé další drobně modifikovat. Do složky **constant** se musí přidat soubor **dynamicMeshDict** [Příloha 9], který popisuje, jaký je zvolen matematický přístup pro pohyb sítě samotné a jakým způsobem se mají deformovat jednotlivé elementy. Pro případ kmitajícího válce v kanálu byla zvolena pevná topologie pohybující se sítě s přístupem označovaným jako **dynamicMotionSolverFvMesh** (odstavec 3.1.1). Matematický model pohybu sítě byl zvolen **velocityLaplacian** (18) a difuzivita **exponential 7 inverseDistance (Gcyl)**. Chování dynamické sítě pro toto nastavení difuzivity je znázorněno na obr. 30. Pro numerickou stabilitu výpočtu je důležitá šikmost a ortogonalita elementů v nejhorším bodě sítě. Právě pro toto nastavení difuzivity byl výpočet numericky stabilní.

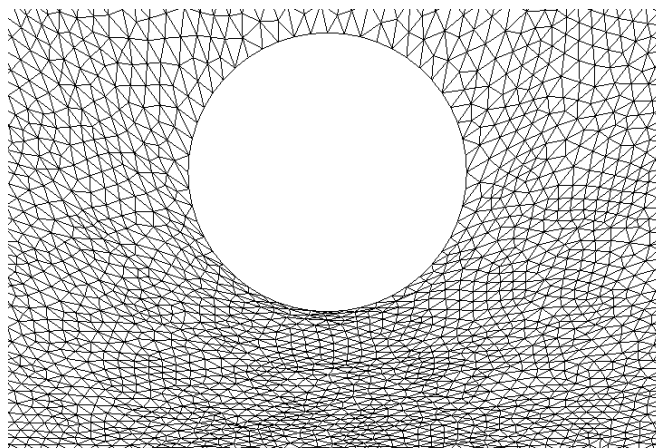
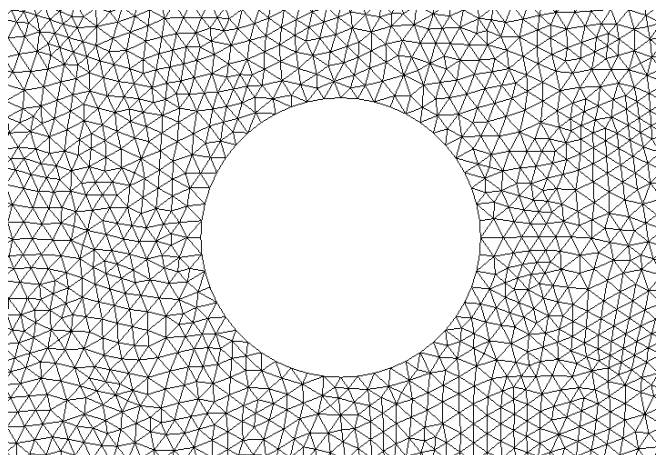
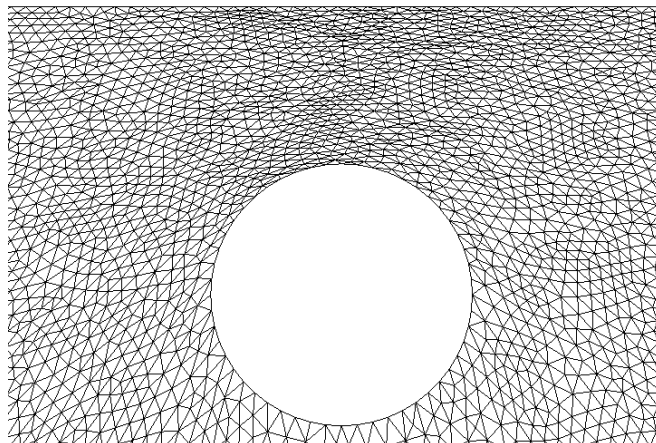
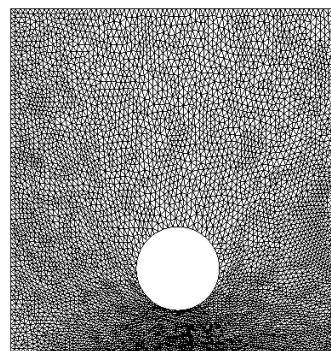
Time: 2.250000s



Time: 2.500000s



Time: 2.750000s



Obr. 30: Deformace dynamické sítě vlivem pohybu válce v kanálu s nastavením: knihovna pro pohyb sítě - *dynamicMotionSolverFvMesh* s matematickým modelem pohybu sítě - *velocityLaplacian* a difuzivitou - *exponential 7 inverseDistance* (Gcyl).

Posledním souborem, který je u výpočtu na dynamické síti navíc oproti výpočtu na statické síti, je soubor **pointMotionU**. Zde se definují okrajové podmínky pro vlastní pohyb hranic. Struktura souboru je uvedena v příloze 10.

Parametry simulace jsou nastaveny následovně. Tekutina proudící v kanálu má kinematickou viskozitu $\nu = 0,001 \text{ m}^2\text{s}^{-1}$. Proudění bylo řešeno jako laminární simulace o délce 3 sekund s časovým krokem 0,001 sekund. Pro numerický výpočet byl použit řešič **pimpleDyMFoam** a pro dynamickou síť byl zvolen model **dynamicMotionSolverFvMesh** s matematickým popisem pohybu sítě **velocityLaplacian** (18) a difuzivitou **exponential 7 inverseDistance** (Gcyl). Okrajové podmínky na hranicích byly deklarovány jako:

$$\text{Gin:} \quad U(0, y, t) = \frac{4 \cdot 1,5 \cdot y(0,41 - y)}{0,41^2} \text{ [m/s]}, \frac{\partial p}{\partial n} = 0 \text{ Pa}, \omega = 0 \text{ rad/s}, A = 0 \text{ m}$$

$$\text{Gout:} \quad \frac{\partial u}{\partial n} = 0 \text{ m/s}, p = 0 \text{ Pa}, \omega = 0 \text{ rad/s}, A = (0, 0, 0) \text{ m}$$

$$\text{Gwall:} \quad u = 0 \text{ m/s}, \frac{\partial p}{\partial n} = 0 \text{ Pa}, \omega = 0 \text{ rad/s}, A = (0, 0, 0) \text{ m}$$

$$\text{Gcyl:} \quad u = \omega, \frac{\partial p}{\partial n} = 0 \text{ Pa}, \omega = 6,283 \text{ rad/s}, A = (0, 0,1, 0) \text{ m}$$

Podrobnější informace o okrajových podmínkách a jejich konkrétním nastavení jsou uvedeny v kapitole 2.2.2. Nicméně tyto podmínky se týkaly především statické sítě, kdy se nám válec v kanálu nepohyboval. Okrajová podmínka spjatá s dynamickou sítí, kdy dochází k pohybu válce v kanálu, je **movingWallVelocity**, která bude použita na hranici samotného válce označenou Gcyl. Tím se zaručí, že rychlost proudění bude přesně na povrchu válce rovna rychlosti, s jakou se válec pohybuje.

Většina použitých okrajových podmínek je již známa z kapitoly 2.2.2. Nicméně se zde nově vyskytuje okrajová podmínka **oscillatingVelocity**, která není standardně v programu OpenFoam implementována, a proto ji bylo nutné do programu doplnit manuálně. Tato okrajová podmínka zaručí harmonický pohyb předepsané hranice (19). Pro řešení pohyblivého válce v kanálu byla tato okrajová podmínka použita na hranici válce (Gcyl). Jednotlivé parametry, které je potřeba nastavit, jsou vcelku logické a vycházejí z obecného popisu harmonické funkce sinus.

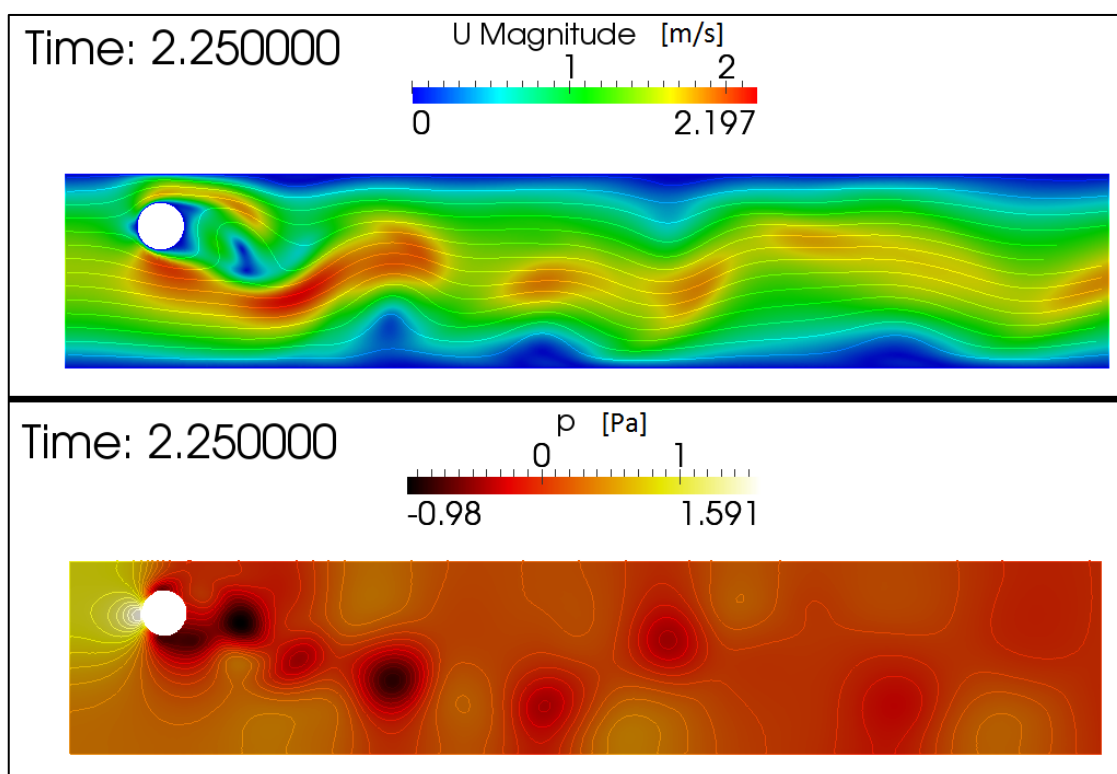
Parametr amplitudy A (*amplitude* (x, y, z))) je definován jako vektor, aby bylo možné definovat směr a velikost pohybu kmitající hranice. U velikosti amplitudy je

potřeba dát si pozor, aby válec nevyjel mimo kanál, tím by došlo samozřejmě k nepřipustné změně topologie geometrie i sítě a výpočet by havaroval.

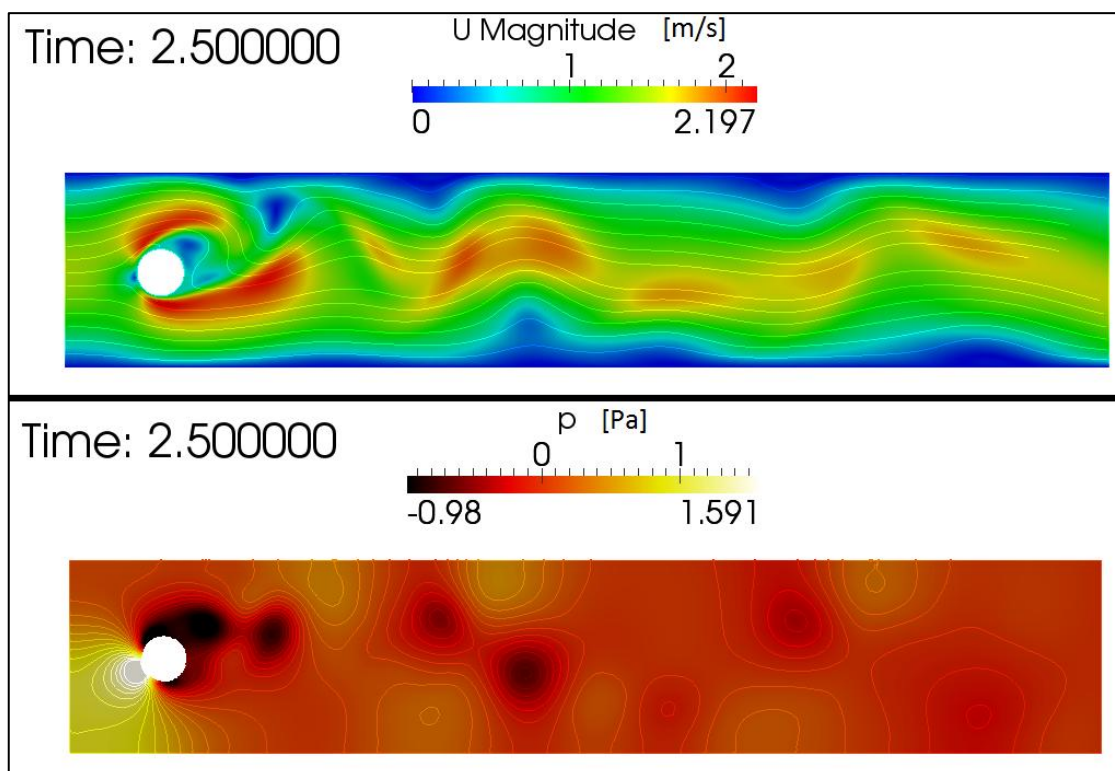
Celý výpočet byl realizován na síti, která měla 190 970 elementů.

3.2.2 Výsledky numerické simulace

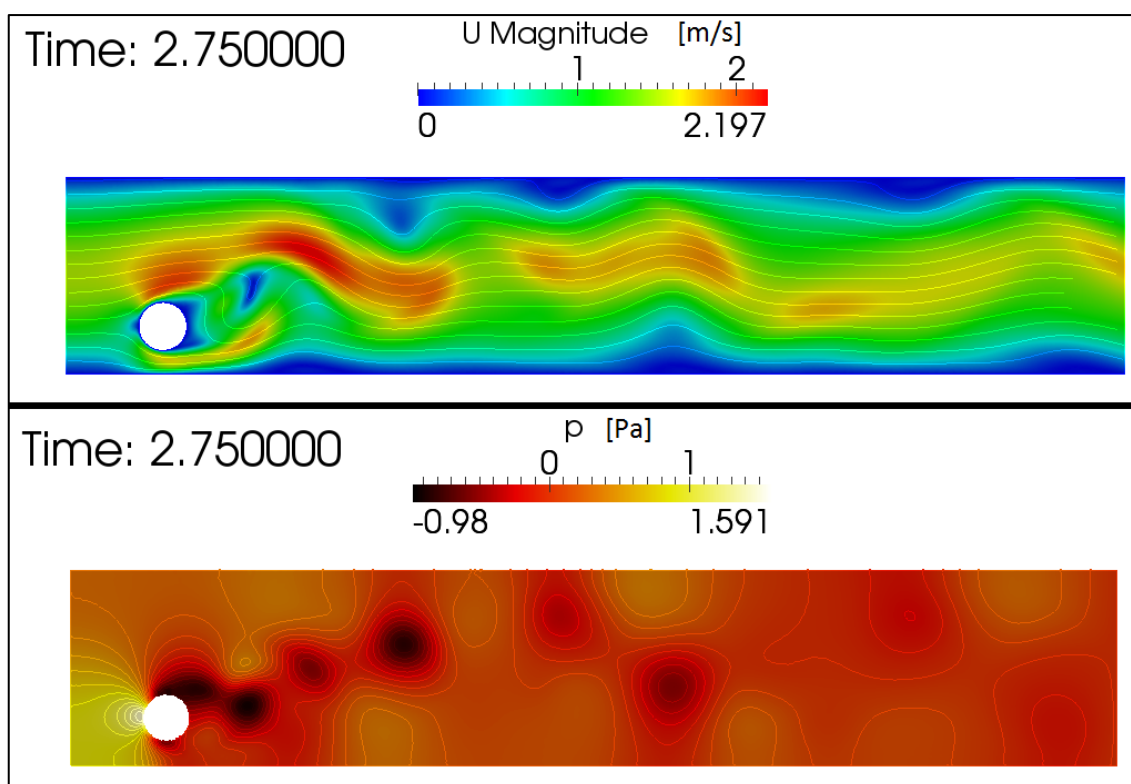
Vizualizace pohybujícího se válce v kanálu je znázorněna na následujících obrázcích, kde na obr. 31 je zobrazen válec v čase 2,25 sekund, což odpovídá vrchní poloze pohybujícího se válce v kanálu. Dalším obrázkem je obr. 32, který znázorňuje válec v čase 2,5 sekund, což odpovídá prostřední (nulové) poloze v kanálu při pohybu válce směrem dolů. Na obrázku (obr. 33) je zobrazen válec v jeho spodní poloze, která odpovídá času 2,75 sekund a na posledním obrázku (obr. 34) je válec ve střední poloze při pohybu nahoru.



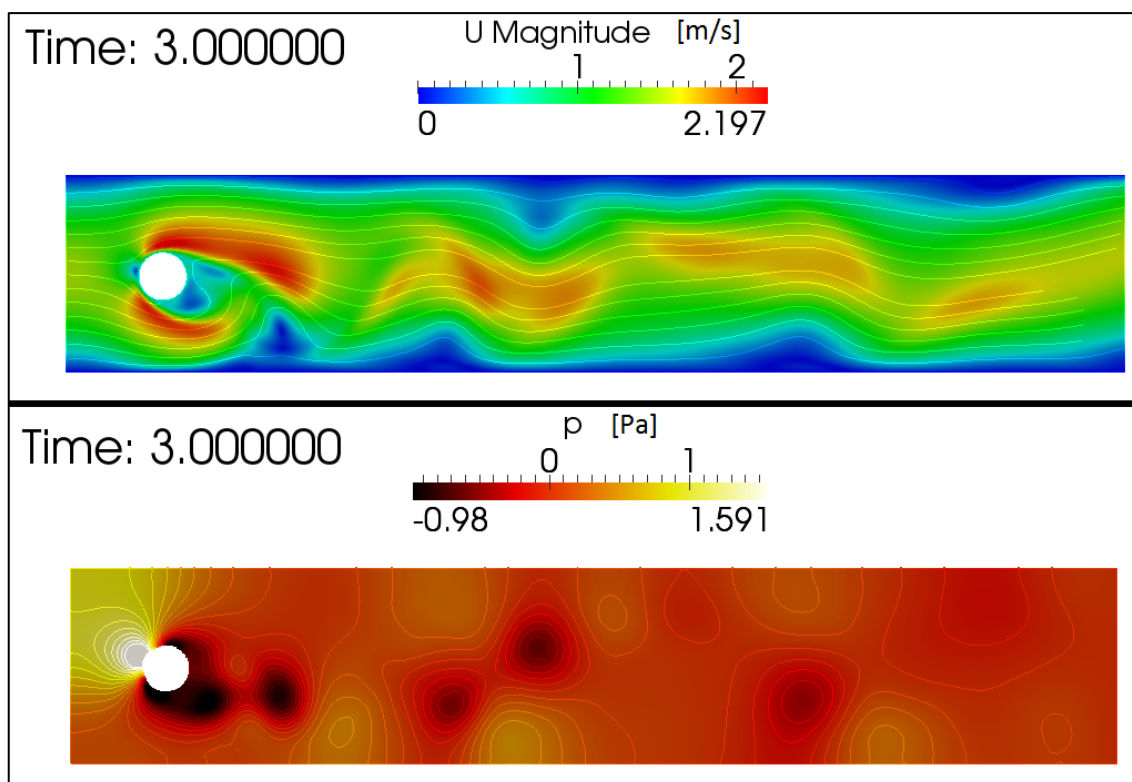
Obr. 31: Válec v kanálu v čase $t = 2.25$ s (horní poloha)



Obr. 32: Válec v kanálu při pohybu dolů v čase $t = 2.5$ s (střední poloha)



Obr. 33: Válec v kanálu v čase $t = 2.75$ s (dolní poloha)



Obr. 34: Válec v kanálu při pohybu nahoru v čase $t = 3$ s (střední poloha)

V každém z výše uvedených obrázků je vždy na vrchním z nich zobrazen průběh rychlosti včetně proudnic. Na spodním obrázku je vždy znázorněn průběh tlaku včetně vykreslení izobar.

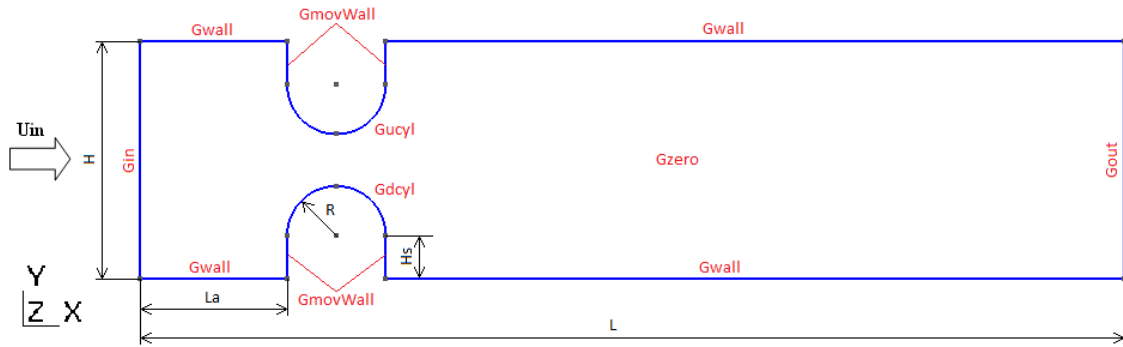
Z výsledků vizualizace je patrné, že se za válcem v kanálu tvoří víry. To je vidět především z průběhu tlaku, kdy uprostřed víru očekáváme nízkou hodnotu tlaku. V našich obrázcích to odpovídá tmavým až černým místům. Stacionární tlak před válcem je větší nežli tlak za válcem, což znamená, že na válec působí síla, která se ho snaží unést ve směru proudění. Průběhy rychlosti a tlaku při horní (obr. 31) a dolní poloze (obr. 33) pohybujícího se válce jsou si vzájemně velmi podobné, jen jsou oproti sobě převrácené, což je ale vcelku logické.

Vzájemně podobné, ale také převrácené jsou vizualizace pohybujícího se válce v kanálu v jeho střední poloze. Záleží, zda se válec pohybuje nahoru (obr. 34) nebo dolů (obr. 32). V těchto dvou případech se v kanálu vyskytují vyšší hodnoty rychlosti a vyšší i nižší hodnoty tlaku oproti vizualizacím, kdy je válec ve své horní či dolní poloze. Samozřejmě i v případech, kdy je válec ve své střední poloze, tak v prostoru kanálu za válcem je patrná tvorba víru, což je opět vidět především ze zobrazeného tlakového průběhu.

3.3 Kmitající profil hlasivek

3.3.1 Zjednodušený model hlasivek

V této kapitole bude rozebrán o něco složitější problém, než je kmitající válec v kanálu, kde se pohybovala pouze jedna část geometrie. Zde se budou pohybovat části dvě a to přesně proti sobě. Bude se jednat o velmi zjednodušený model lidských hlasivek, jejichž geometrie je zobrazena na obr. 35.



Obr. 35: Geometrie jednoduchého profilu hlasivek včetně pojmenování hranic

Rozměry zjednodušených hlasivek jsou: $H = 0,012$ m, $L = 0,05$ m, $R = 0,0025$ m, $La = L/5 - R = 0,01 - 0,0025 = 0,0075$ m a $Hs = H/5,5 = 0,002182$ m.

Podrobnější popis možností práce s pohyblivou sítí včetně vhodných okrajových podmínek byl uveden v kapitolách 3.1 a 3.2. Proto zde budou zmíněna pouze konkrétní nastavení bez bližších komentářů.

Parametry simulace jsou nastaveny následovně. Vzduch proudící v hlasivkách má kinematickou viskozitu $\nu = 1,58 \cdot 10^{-5} \text{ m}^2 \text{ s}^{-1}$. Proudění bylo řešeno jako laminární simulace o délce 0,06 sekund s časovým krokem $5 \cdot 10^{-6}$ sekund. Pro numerický výpočet byl použit řešič pimpleDyMFoam a pro dynamickou síť byl zvolen model dynamicMotionSolverFvMesh s matematickým popisem pohybu sítě velocityLaplacian (18) a difuzivitou exponential 800 inverseDistance (Gdcyl, Gucyl). Výpočet byl realizován na síti, která měla 96294 elementů. Okrajové podmínky na hranicích byly deklarovány jako:

$$\text{Gin:} \quad u = 1 \text{ ms}^{-1}, \frac{\partial p}{\partial n} = 0 \text{ Pa}, \omega = 0 \text{ rad/s}, A = (0, 0, 0) \text{ m}$$

$$\text{Gout:} \quad \frac{\partial u}{\partial n} = 0 \text{ ms}^{-1}, p = 0 \text{ Pa}, \omega = 0 \text{ rad/s}, A = (0, 0, 0) \text{ m}$$

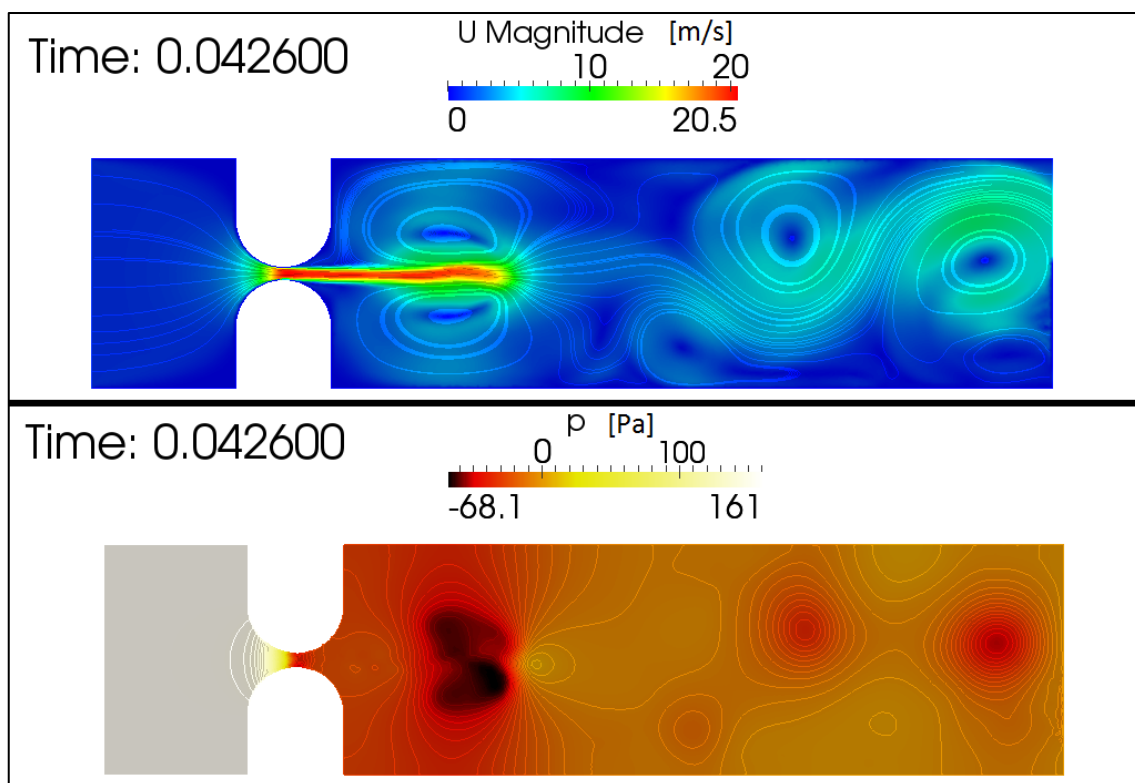
$$\text{Gwall:} \quad u = 0 \text{ ms}^{-1}, \frac{\partial p}{\partial n} = 0 \text{ Pa}, \omega = 0 \text{ rad/s}, A = (0, 0, 0) \text{ m}$$

GmovWall: $u = \omega, \frac{\partial p}{\partial n} = 0 \text{ Pa}, \frac{\partial \omega}{\partial n} = 0, \omega \mathbf{n} = 0$

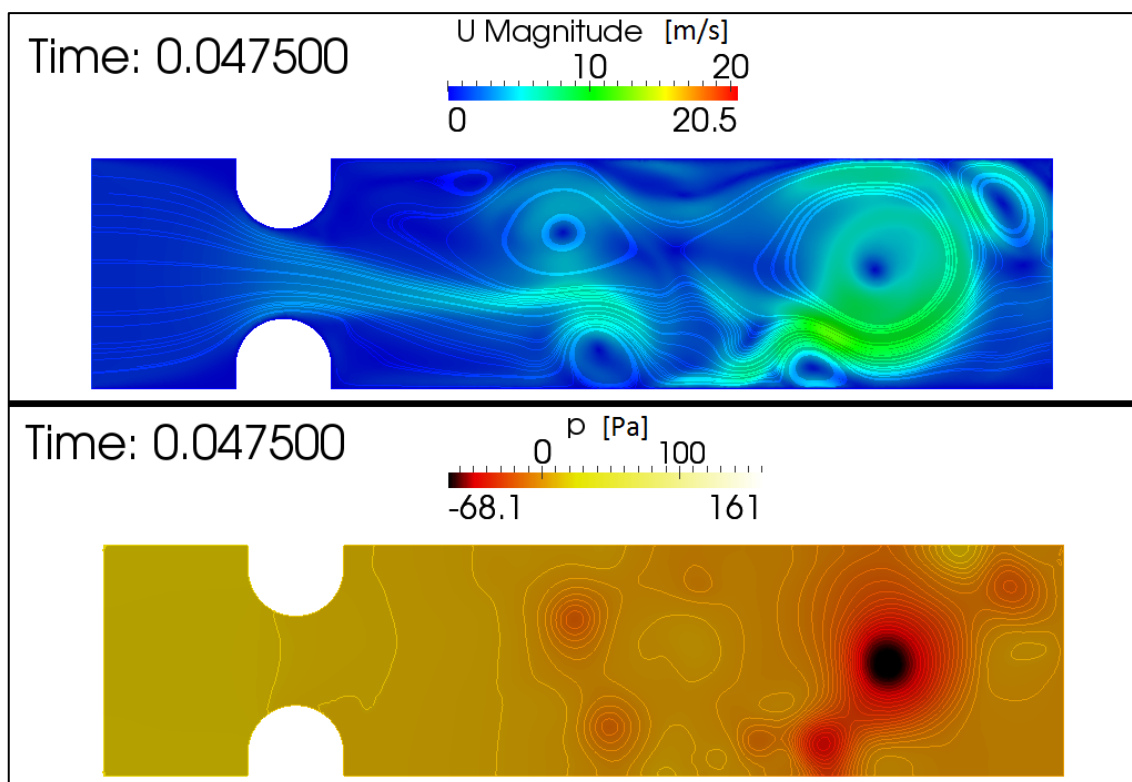
Gucyl: $u = \omega, \frac{\partial p}{\partial n} = 0 \text{ Pa}, \omega = 628.319 \text{ rad/s}, A = (0, -0.001, 0) \text{ m}$

Gdcyl: $u = \omega, \frac{\partial p}{\partial n} = 0 \text{ Pa}, \omega = 628.319 \text{ rad/s}, A = (0, 0.001, 0) \text{ m}$

Výsledky vizualizace jsou vidět na obr. 36 a obr. 37.



Obr. 36: Průběh rychlosti a tlaku při přivření hlasivek

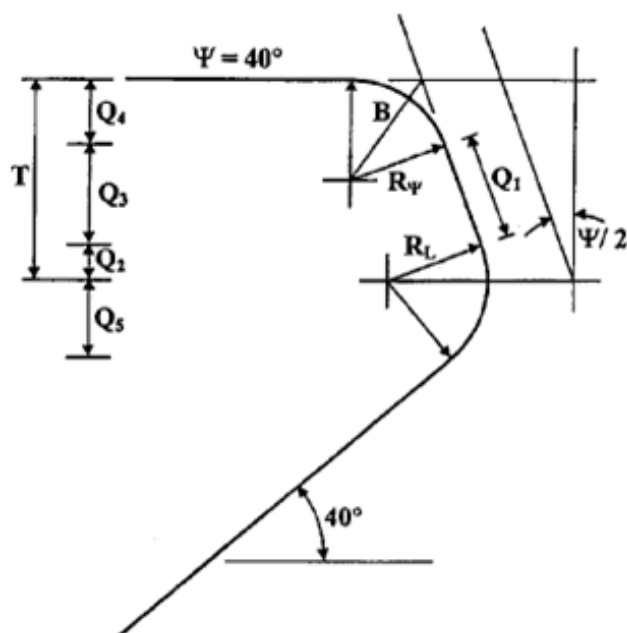


Obr. 37: Průběh rychlosti a tlaku při otevřených hlasivkách

Z průběhů je patrné, že charakter proudění je významněji ovlivněn při skoro uzavřených hlasivkách (obr. 36), nežli při hlasivkách otevřených (obr. 37). Tudíž je zajímavější zkoumat chování proudění v téměř uzavřených hlasivkách, kde je vidět, že před místem zúžení je hodnota tlaku o mnoho větší, nežli za zúženým místem. Naopak v místě zúžení je hodnota tlaku minimální. Z rychlostního průběhu je vidět, že v místě zúžení je hodnota rychlosti maximální a v prostoru za zúžením hlasivek se tvoří víry.

3.3.2 Model hlasivek M5

Realitě bližší model hlasivek nese označení M5, kterým se zabývá práce [14]. Jedná se o standardní tvar pro modelování proudění v lidských hlasivkách. Oproti předchozímu případu má model M5 složitější geometrii (obr. 38).



Obr. 38: Detailní popis geometrie hlasivky. Převzato z práce [14].

Základní rozměry hlasivky M5 jsou:

$$R_0 = 0,0987 \text{ cm}$$

$$T = 0,3 \text{ cm}$$

$$-40^\circ \leq \psi \leq 40^\circ$$

Rovnice popisující geometrii výčnělku hlasivky M5 jsou: [14]

$$R_\psi = \frac{R_0}{1 - \sin\left(\frac{\psi}{2}\right)} \quad (20)$$

$$R_L = R_{40} = \frac{T}{2} \quad (21)$$

$$B = \frac{\sqrt{2}R_\psi}{\sqrt{1 + \sin\left(\frac{\psi}{2}\right)}} = \frac{R_0 \sec\left(\frac{\psi}{2}\right)}{\sqrt{(1 - \sin\left(\frac{\psi}{2}\right))/2}} \quad (22)$$

$$\begin{aligned} Q_1 &= (T - R_\psi) \sec\left(\frac{\psi}{2}\right) + (R_\psi - R_L) \tan\left(\frac{\psi}{2}\right) = \\ &= \left(T - R_0 - R_L \sin\left(\frac{\psi}{2}\right)\right) \sec\left(\frac{\psi}{2}\right) \end{aligned} \quad (23)$$

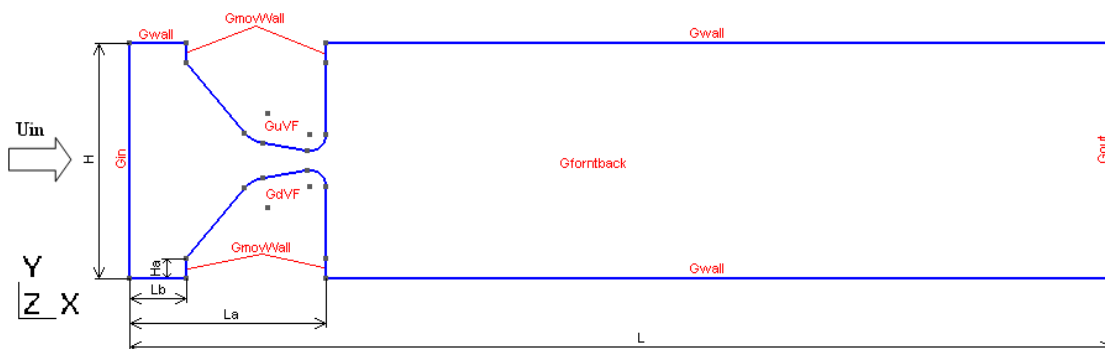
$$Q_2 = R_L \sin\left(\frac{\psi}{2}\right) \quad (24)$$

$$Q_3 = Q_1 \cos\left(\frac{\psi}{2}\right) \quad (25)$$

$$Q_4 = R_0 \quad (26)$$

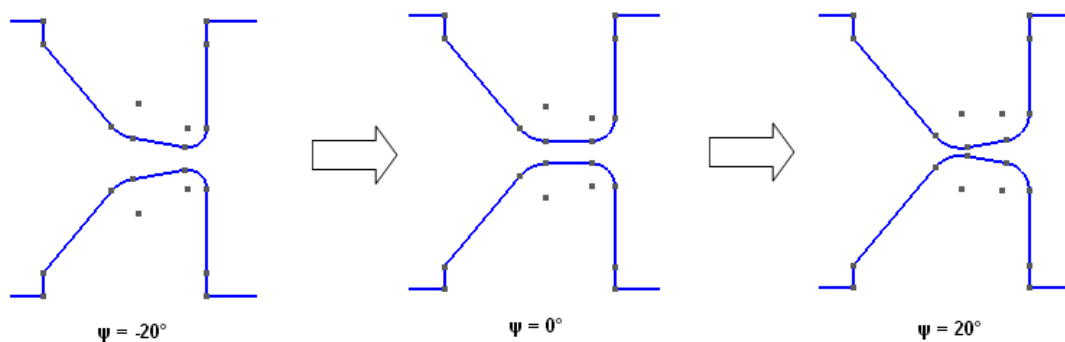
$$Q_5 = R_L \sin(50^\circ) \quad (27)$$

Geometrie modelu hlasivek M5 má parametry $L = 0,05\text{m}$, $L_a = 0,01\text{m}$, $L_b = 0,0072\text{m}$, $H = 0,012\text{m}$ a $H_a = 0,001\text{m}$.

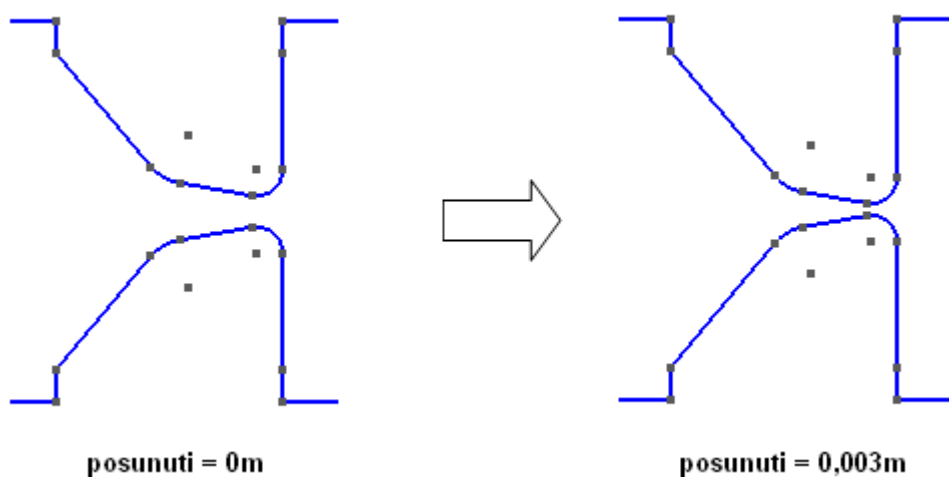


Obr. 39: Geometrie hlasivek M5 včetně označení hranic

Geometrie M5 je parametrizována úhlem ψ , kterým se mění rozložení horní části hlasivky (obr. 40). Druhým parametrem můžeme ovlivňovat celkovou výšku hlasivky (obr. 41).



Obr. 40: Vliv parametru ψ na rozložení vrchní části hlasivky



Obr. 41: Vliv parametru pro posunutí celého hlasivky ($\psi = -20^\circ$)

Parametry simulace jsou nastaveny následovně. Vzduch proudící v hlasívkách má kinematickou viskozitu $\nu = 1.58 \cdot 10^{-5} \text{ m}^2 \cdot \text{s}^{-1}$. Proudění bylo řešeno jako laminární simulace o délce 0,06 sekund s časovým krokem $5 \cdot 10^{-6}$ sekund. Pro numerický výpočet byl použit řešič pimpleDyMFOam a pro dynamickou síť byl zvolen model dynamicMotionSolverFvMesh s matematickým popisem pohybu sítě velocityLaplacian (18) a difuzivitou exponential 1000 inverseDistance (GdVF, GuVF). Výpočet byl realizován na nestrukturované síti, která měla 109 450 elementů. Okrajové podmínky na hranicích byly deklarovány jako:

$$\text{Gin:} \quad u = 1 \text{ ms}^{-1}, \frac{\partial p}{\partial n} = 0 \text{ Pa}, \omega = 0 \text{ rad/s}, A = (0, 0, 0) \text{ m}$$

$$\text{Gout:} \quad \frac{\partial u}{\partial n} = 0 \text{ ms}^{-1}, p = 0 \text{ Pa}, \omega = 0 \text{ rad/s}, A = (0, 0, 0) \text{ m}$$

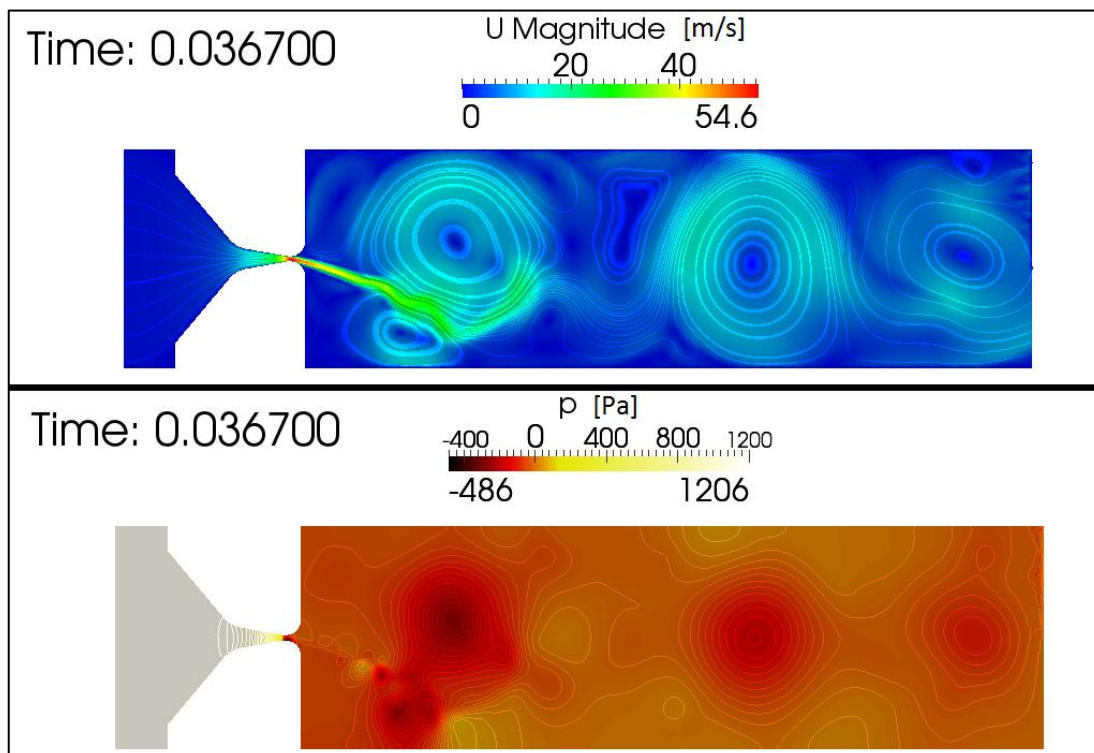
$$\text{Gwall:} \quad u = 0 \text{ ms}^{-1}, \frac{\partial p}{\partial n} = 0 \text{ Pa}, \omega = 0 \text{ rad/s}, A = (0, 0, 0) \text{ m}$$

$$\text{GmovWall:} \quad u = \omega, \frac{\partial p}{\partial n} = 0 \text{ Pa}, \frac{\partial \omega}{\partial n} = 0, \omega \mathbf{n} = 0$$

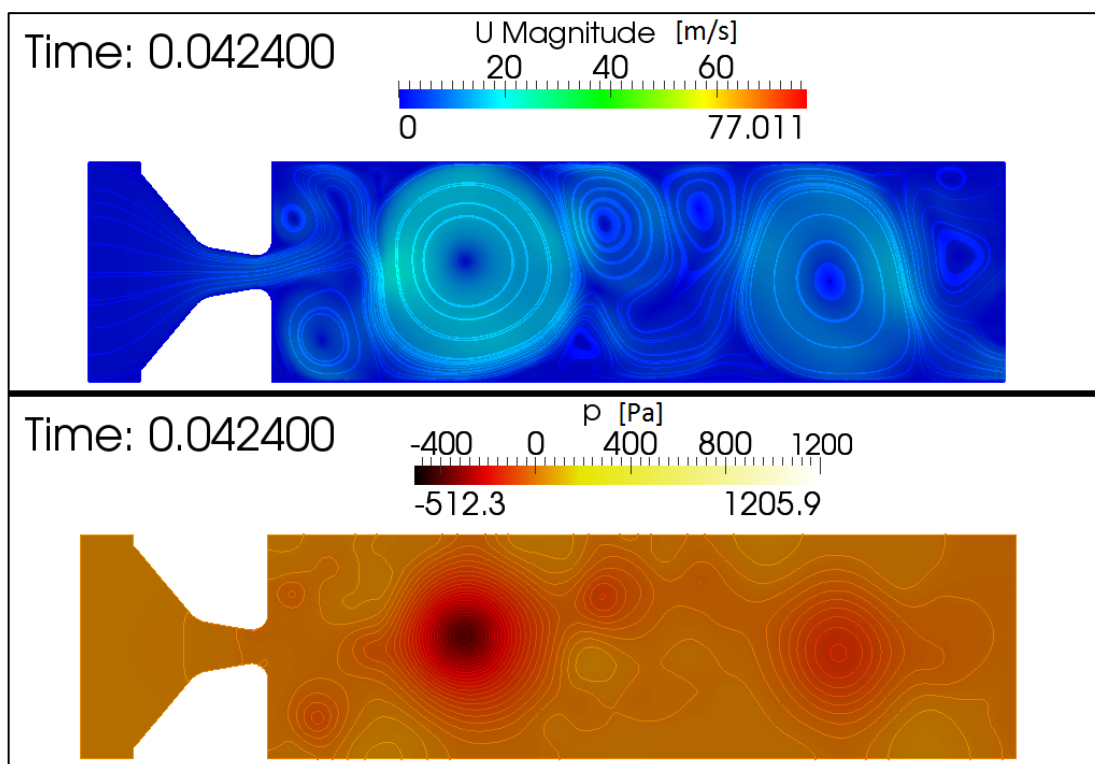
$$\text{GuVF:} \quad u = \omega, \frac{\partial p}{\partial n} = 0 \text{ Pa}, \omega = 628.319 \text{ rad/s}, A = (0, 0.0004, 0) \text{ m}$$

$$\text{GdVF:} \quad u = \omega, \frac{\partial p}{\partial n} = 0 \text{ Pa}, \omega = 628.319 \text{ rad/s}, A = (0, -0.0004, 0) \text{ m}$$

Výsledek následné vizualizace je vidět na obr. 42 a obr. 43.



Obr. 42: Průběh rychlosti a tlaku v uzavřených hlasívkách typu M5 ($\psi = -20^\circ$)



Obr. 43: Průběh rychlosti a tlaku v otevřených hlasivkách typu M5 ($\psi = -20^\circ$)

Za zúženým profilem je vidět, že se tvoří víry. Při uzavření hlasivek je v nejužším místě mezi výčnělky nejvyšší rychlost proudění tekutiny. V případě uzavřených hlasivek je také vidět, že v prostoru před zúžením je vyšší hodnota tlaku nežli v prostoru za zúžením.

4 Závěr

V diplomové práci byl řešen problém numerické simulace obtékání, jak statické, tak i dynamické překážky. Jednalo se o válec v uzavřeném kanálu a o geometrii lidských hlasivek.

V úvodu byla rozebrána fyzika proudění v uzavřených kanálech, včetně základních veličin charakterizujících proudění. Další část diplomové práce byla věnována numerické simulaci obtékání statické překážky. Prvním krokem bylo potřeba stanovit optimální počet elementů sítě, aby výsledky simulace byly věrohodné, ale aby nedocházelo k přílišné zátěži počítače. Druhým krokem bylo pomocí verifikace porovnat výsledky mé simulace s výsledky testovací úlohy [9]. Poslední částí diplomové práce byla numerická simulace obtékání periodicky kmitající překážky. Bylo potřeba určit, jakým způsobem se bude dynamická síť deformovat. K tomu sloužil parametr difuzivita.

Cílem diplomové práce bylo provést numerickou simulaci obtékání statického tělesa, kterým byl válec v kanálu, včetně verifikace výsledků. Optimální počet elementů sítě byl stanoven na cca 190 tisíc. Při verifikaci došlo k porovnávání tlakové difference, Strouhalova čísla, odporového součinitele a součinitele vztlaku. Porovnáním výsledků mé simulace s výsledky simulací uvedených v práci, zabývající se testovací úlohou [9], bylo zjištěno, že se výsledky pouze minimálně odchylují. Chyby výsledků se pohybovaly do 6 %, což lze považovat za vyhovující. Dalším porovnávaným parametrem byly průběhy rychlostních profilů v různých částech kanálu, které byly v práci [9] zjišťovány experimentálně. Rychlostní profily byly porovnány pouze vizuálně a zjistilo se, že se od sebe pouze minimálně odchylují. Při následné vizualizaci byly v prostoru za válcem vidět víry, což také odpovídá teoretickým předpokladům.

V případě dynamické překážky bylo hlavním cílem seznámit se s metodami a přístupy pro řešení pohyblivé a deformující se sítě. Celkem byly provedeny simulace na třech různých geometriích, kterými byly pohybující se válec v kanálu, triviální model kmitajících hlasivek a kmitající model hlasivek M5 [14]. Parametr difuzivita byl pro každý případ nastaven experimentálně. Při vizualizaci bylo patrné odtrhávání vírů za kmitající překážkou a případný nárůst rychlosti proudění tekutiny při zúženém profilu.

Geometrie a síť modelovaných problémů byly vytvořeny pomocí programu Gmsh. Pro vlastní výpočet byl použit výpočetní software OpenFoam a vizualizace byla

provedena programem ParaView. Všechny použité softwary jsou OpenSource, což je zcela jistě výhoda. Jsou zdarma a lze si zdrojové kódy dle libosti upravovat. V případě OpenFoamu je zdrojový kód psán pomocí programovacího jazyka C++. Ve všech simulacích byl použit pro výpočet proudění laminární model.

Modelování dynamických sítí je v současné době předmětem intenzivního výzkumu. Příkladem jejich použití může být modelování proudění kolem kmitajících lan mostních konstrukcí, proudění kolem vysokých budov, obtékání reaktorových tyčí chladičem, proudění kolem leteckých profilů, apod.. Právě na případ modelování dynamických sítí lze navázat v další práci, kde bude potřeba vytvořit, případně upravit výpočetní knihovny OpenFoamu tak, aby odpovídaly zkoumané problematice.

Použitá literatura

- [1] *OpenFoam – The Open Source CFD Toolbox – User guide, Version 1.7.1*; 25th August 2010 [online]. [cit. 2011-09-02].
URL: < <http://foam.sourceforge.net/docs/Guides-a4/UserGuide.pdf> >
- [2] Moreland Kenneth; *The ParaView Tutorial Version 3.8*; [online]. [cit. 2011-06-02].
URL: < <http://paraview.org/Wiki/images/8/88/ParaViewTutorial38.pdf> >
- [3] Christophe Geuzaine and Jean-Fran,cois Remacle; *Gmsh GUI Tutorial I: How to Create a Simple 2D Model?*; 4th January 2006 [online]. [cit. 2011-18-02].
URL: <http://ffep.sourceforge.net/Download/gui_tutorial.pdf>
- [4] Munson, Young, Okiishi, Huebsch. *Fundamentals of Fluid Mechanics – Sixth Edition*. 2010 John Wiley & Sons (Asia). 723 stran.
- [5] M. E. Young, A. Ooi, *Comparative Assessment of LES and URANS for Flow Over a Cylinder at a Reynolds Number of 3900*; 2nd December 2007 [online]. [cit. 2011-10-04].
URL: <http://espace.library.uq.edu.au/eserv/UQ:121281/Young_afmc_16_07.pdf>
- [6] Tabulky kinematické viskozity
URL: < <http://www.converter.cz/tabulky/kinematicka-viskozita.htm> >
[cit. 2011-05-02].
- [7] Vírová řada
URL: < http://www.simerics.com/gallery_vortex.html >
[cit. 2011-05-02].
- [8] Parabolická okrajová podmínka
URL: < <http://engineeringskills.wikidot.com/concepts> >
[cit. 2011-09-02].
- [9] M. Schäfer and S. Turek; *Benchmark Computations of Laminar Flow Around a Cylinder*; 1996

- [10] Válec v kanálu
URL: <http://nptel.iitm.ac.in/courses/Webcourse-contents/IIT-KANPUR/FLUID-MECHANICS/lecture-31/31-3_mechanics.htm>
[cit. 2011-08-02].
- [11] H K Versteeg, W Malalasekera. *An Introduction to Computational Fluid Dynamics – Second Edition*. 2007 Pearson Education Limited (England). 503 stran.
- [12] Frank M. White. *Fluid Mechanics – Sixth Edition*. 2008 McGraw-Hill Higher Education (USA). 864 stran.
- [13] Pirooz Moradnia. *A tutorial on how to use Dynamic Mesh solver IcoDyMFoam*; 2008 [online]. [cit. 2011-02-04]. URL:<http://www.tfd.chalmers.se/~hani/kurser/OS_CFD_2007/PiroozMoradnia/OpenFOAM-rapport.pdf>
- [14] Ronald C. Scherer, Daoud Shinwari. *Intraglottal pressure profiles for a symmetric and oblique glottis with a divergence angle of 10 degrees*. 2001 Journal of the Acoustical Society of America. Str. 1616 - 1630
- [15] Viskozita
URL:<http://www.vscht.cz/met/stranky/vyuka/labcv/labor/res_stanoveni_viskozity_roztoku/teorie.htm>
[cit. 2011-15-04].
- [16] Christophe Kassiotis. *Which strategy to move the mesh in the Computational Fluid Dynamic code OpenFOAM*; 12.04. 2008 [online]. [cit. 2011-17-04].
URL: < <http://perso.crans.org/kassiotis/openfoam/movingmesh.pdf>>

Přílohy

Soubor boundary: popis hranic objektu

Příloha 1: soubor boundary

```
/*-----*- C++ -*-----*\
| ===== |
| \\      / F ield      | OpenFOAM: The Open Source CFD Toolbox |
| \\      / O peration  | Version: 1.7.1 |
| \\      / A nd        | Web: www.OpenFOAM.com |
|  \\     M anipulation |
|-----*\
FoamFile
{
    version      2.0;
    format        ascii;
    class         polyBoundaryMesh;
    location      "constant/polyMesh";
    object        boundary;
}
// *****

5
(
    Gzero
    {
        type            empty;
        nFaces          63472;
        startFace       47261;
    }
    Gwall
    {
        type            patch;
        nFaces          548;
        startFace       110733;
    }
    Gout
    {
        type            patch;
        nFaces          51;
        startFace       111281;
    }
    Gin
    {
        type            patch;
        nFaces          51;
        startFace       111332;
    }
    Gcyl
    {
        type            patch;
        nFaces          36;
        startFace       111383;
    }
)
// *****
```


Soubor turbulenceProperties: charakter proudění

Příloha 2: soubor turbulenceProperties

```
/*-----* C++ *-----*\
| ===== |
| \ \ / F i e l d | OpenFOAM: The Open Source CFD Toolbox |
| \ \ / O p e r a t i o n | Version: 1.7.1 |
| \ \ / A n d | Web: www.OpenFOAM.org |
| \ \ M a n i p u l a t i o n |
\*-----*/
FoamFile
{
    version      2.0;
    format       ascii;
    class        dictionary;
    location     "constant";
    object       turbulenceProperties;
}
// * * * * *

simulationType laminar;

// * * * * *
```

Soubor transportProperties: fyzikální konstanty média

Příloha 3: soubor transportProperties

```
/*-----*- C++ -*------*\
| ===== |
| \\ / F i e l d | OpenFOAM: The Open Source CFD Toolbox |
| \\ / O p e r a t i o n | Version: 1.7.1 |
| \\ / A n d | Web: www.OpenFOAM.org |
| \\ M a n i p u l a t i o n |
\*-----*/
FoamFile
{
    version      2.0;
    format       ascii;
    class        dictionary;
    location     "constant";
    object       transportProperties;
}
// *****

transportModel Newtonian;

nu              nu [ 0 2 -1 0 0 0 0 ] 0.001;

// *****
```

Soubor fvSchemes: numerická schémata metody konečných objemů

Příloha 4: soubor fvSchemes

```
/*----- C++ -----*\
| ===== |
| \\ / F i e l d | OpenFOAM: The Open Source CFD Toolbox |
| \\ / O peration | Version: 1.7.1 |
| \\ / A nd | Web: www.OpenFOAM.org |
| \\ M anipulation |
\*-----*/

FoamFile
{
    version      2.0;
    format       ascii;
    class        dictionary;
    location     "system";
    object       fvSchemes;
}

// *****

ddtSchemes
{
    default      Euler;
}

gradSchemes
{
    default      Gauss linear;
    grad(p)      Gauss linear;
    grad(U)      Gauss linear;
}

divSchemes
{
    default      none;
    div(phi,U)   Gauss limitedLinearV 1;
    div(phi,k)   Gauss limitedLinear 1;
    div(phi,epsilon) Gauss limitedLinear 1;
    div(phi,R)   Gauss limitedLinear 1;
    div(R)       Gauss linear;
    div(phi,nuTilda) Gauss limitedLinear 1;
    div((nuEff*dev(grad(U).T()))) Gauss linear;
}

laplacianSchemes
{
    default      none;
    laplacian(nuEff,U) Gauss linear corrected;
    laplacian((1|A(U)),p) Gauss linear corrected;
    laplacian(DkEff,k) Gauss linear corrected;
    laplacian(DepsilonEff,epsilon) Gauss linear corrected;
    laplacian(DREff,R) Gauss linear corrected;
    laplacian(DnuTildaEff,nuTilda) Gauss linear corrected;
    laplacian(diffusivity,cellMotionU) Gauss linear uncorrected;
    laplacian(rAU,p) Gauss linear corrected;
    laplacian(rAU,pcorr) Gauss linear corrected;
}
```

```

interpolationSchemes
{
    default          linear;
    interpolate(U)    linear;
}

snGradSchemes
{
    default          corrected;
}

fluxRequired
{
    default          no;
    pcorr            ;
    p                 ;
}

// *****

```

Soubor fvSolution: lineární řešiče

Příloha 5: soubor fvSolution

```
/*----- C++ -----*\
| ===== |
| \\ / F i e l d | OpenFOAM: The Open Source CFD Toolbox |
| \\ / O p e r a t i o n | Version: 1.7.1 |
| \\ / A n d | Web: www.OpenFOAM.org |
| \\ M a n i p u l a t i o n |
\*-----*/

FoamFile
{
    version      2.0;
    format       ascii;
    class        dictionary;
    location     "system";
    object       fvSolution;
}

// *****

solvers
{
    pcorr
    {
        solver      GAMG;
        tolerance    1e-06;
        relTol       0;
        smoother     GaussSeidel;
        cacheAgglomeration true;
        nCellsInCoarsestLevel 10;
        agglomerator  faceAreaPair;
        mergeLevels   1;
    }

    p
    {
        solver      GAMG;
        tolerance    1e-06;
        relTol       0.01;
        smoother     GaussSeidel;
        cacheAgglomeration true;
        nCellsInCoarsestLevel 10;
        agglomerator  faceAreaPair;
        mergeLevels   1;
    }

    pFinal
    {
        solver      GAMG;
        tolerance    1e-06;
        relTol       0;
        smoother     GaussSeidel;
        cacheAgglomeration true;
        nCellsInCoarsestLevel 10;
        agglomerator  faceAreaPair;
        mergeLevels   1;
    }
}
```

```

U
{
    solver          PBiCG;
    preconditioner   DILU;
    tolerance        1e-05;
    relTol           0.1;
}

UFinal
{
    solver          PBiCG;
    preconditioner   DILU;
    tolerance        1e-05;
    relTol           0;
}

k
{
    solver          PBiCG;
    preconditioner   DILU;
    tolerance        1e-05;
    relTol           0;
}

epsilon
{
    solver          PBiCG;
    preconditioner   DILU;
    tolerance        1e-05;
    relTol           0;
}

cellMotionU
{
    solver          PCG;
    preconditioner   DIC;
    tolerance        1e-08;
    relTol           0;
}

}

PIMPLE
{
    nOuterCorrectors 2;
    nCorrectors       2;
    nNonOrthogonalCorrectors 0;
    pRefCell           0;
    pRefValue           0;
}

relaxationFactors
{
    U          1;
    k          1;
    epsilon    1;
}

// *****

```

Soubor controlDict: parametry simulace

Příloha 6: soubor controlDict

```
/*-----* C++ -*-----*\
|=====|
| \\ / F i e l d | OpenFOAM: The Open Source CFD Toolbox |
| \\ / O p e r a t i o n | Version: 1.7.1 |
| \\ / A n d | Web: www.OpenFOAM.org |
| \\ M a n i p u l a t i o n |
\*-----*/

FoamFile
{
    version      2.0;
    format       ascii;
    class        dictionary;
    location     "system";
    object       controlDict;
}

// *****

application      pimpleFoam;
startFrom        startTime;
startTime        0;
stopAt           endTime;
endTime          6;
deltaT           0.008;
writeControl      adjustableRunTime;
writeInterval     0.01;
purgeWrite        0;
writeFormat       ascii;
writePrecision    6;
writeCompression  compressed;
timeFormat        general;
timePrecision     6;
runTimeModifiable yes;
adjustTimeStep    yes;
maxCo             1;
libs ("libmyBCs.so");

functions
{
    probes
    {
        type      probes;
        // where to load it from
        functionObjectLibs ( "libsampling.so" );
        // Name of the directory for probe data
        name      probes;
        probeLocations
        (
            ( 0.1 0 0.04 )
        );
        // Fields to be probed
        fields      ( p u );
        // Write at same frequency as fields
        outputControl    outputTime;
        outputInterval    1;
    }
}
```

```

forces
{
    type                forces;
    functionObjectLibs  ( "libforces.so" );
    outputControl        outputTime;
    outputInterval      1;
    patches              (Gcyl);
    pName                p;
    UName                U;
    rhoName              rhoInf;
    log                  true;
    rhoInf               1;
    CofR                 (0 0 0);
}
}

```

```

// *****

```


Soubor U: okrajové podmínky pro rychlost

Příloha 7: soubor U

```
/*-----*- C++ -*-----*\
| =====|
| \\      / F i e l d      | OpenFOAM: The Open Source CFD Toolbox |
| \\      / O peration     | Version: 1.7.1                        |
| \\      / A nd           | Web:      http://www.OpenFOAM.org      |
|  \\     / M anipulation  |                                     |
\*-----*-*/

FoamFile
{
    version      2.0;
    format       ascii;
    class        volVectorField;
    object       U;
}

// *****

dimensions      [0 1 -1 0 0 0 0];

internalField    uniform (0 0 0);

boundaryField
{
    Gin
    {
        type      parabolicVelocity;
        n         (1 0 0);
        y         (0 1 0);
        maxValue   1.5;
        value      (0 0 0);
    }

    Gout
    {
        type      zeroGradient;
    }

    Gwall
    {
        type      fixedValue;
        value      uniform (0 0 0);
    }

    Gcyl
    {
        type      fixedValue;
        value      uniform (0 0 0);
    }

    Gzero
    {
        type      empty;
    }
}

// *****
```

Soubor p: okrajové podmínky pro tlak

Příloha 8: soubor p

```
/*-----* C++ *-----*\
| ===== |
| \\      / F i e l d | OpenFOAM: The Open Source CFD Toolbox |
| \\      / O p e r a t i o n | Version: 1.7.1 |
| \\      / A n d | Web: http://www.OpenFOAM.org |
|  \\     / M a n i p u l a t i o n |
\*-----*/
FoamFile
{
    version      2.0;
    format       ascii;
    class        volScalarField;
    object       p;
}
// *****

dimensions      [0 2 -2 0 0 0 0];

internalField    uniform 0;

boundaryField
{
    Gin
    {
        type      zeroGradient;
    }

    Gout
    {
        type      fixedValue;
        value      uniform 0;
    }

    Gwall
    {
        type      zeroGradient;
    }

    Gcyl
    {
        type      zeroGradient;
    }

    Gzero
    {
        type      empty;
    }
}
// *****
```

Soubor dynamicMeshDict: deformace sítě - difusivita

Příloha 9: soubor dynamicMeshDict

```
/*-----* C++ *-----*\
|=====|
| \\ / F i e l d | OpenFOAM: The Open Source CFD Toolbox |
| \\ / O peration | Version: 1.7.1 |
| \\ / A nd | Web: www.OpenFOAM.org |
| \\ M anipulation |
\*-----*/

FoamFile
{
    version      2.0;
    format       binary;
    class        dictionary;
    location     "constant";
    object       dynamicMeshDict;
}

// *****

dynamicFvMesh      dynamicMotionSolverFvMesh;

motionSolverLibs   ( "libfvMotionSolvers.so" );

solver             velocityLaplacian;

diffusivity        exponential 7 inverseDistance (Gcyl);

// *****
```

Soubor pointMotionU: okrajové podmínky pro pohyb sítě

Příloha 10: soubor pointMotionU

```

/*-----*- C++ -*-----*\
| =====|
| \ \ / F i e l d | OpenFOAM: The Open Source CFD Toolbox |
| \ \ / O p e r a t i o n | Version: 1.7.1 |
| \ \ / A n d | Web: http://www.OpenFOAM.org |
| \ \ M a n i p u l a t i o n |
|-----*\
FoamFile
{
    version      2.0;
    format       ascii;
    class        pointVectorField;
    object       pointMotionU;
}
// *****

dimensions      [0 1 -1 0 0 0];

internalField    uniform (0 0 0);

boundaryField
{
    Gcyl
    {
        type      oscillatingVelocity;
        omega      6.283; // 1Hz
        amplitude   (0 0.1 0);
        value       uniform (0 0 0);
    }

    Gwall
    {
        type      fixedValue;
        value      uniform (0 0 0);
    }
    Gin
    {
        type      fixedValue;
        value      uniform (0 0 0);
    }

    Gout
    {
        type      fixedValue;
        value      uniform (0 0 0);
    }
}

// *****

```